Winter 1-15-2021

Deep Learning for Task-Based Image Quality Assessment in Medical Imaging

Weimin Zhou
Washington University in St. Louis

Recommended Citation

Zhou, Weimin, "Deep Learning for Task-Based Image Quality Assessment in Medical Imaging" (2021). *Engineering and Applied Science Theses & Dissertations*. 602.
https://openscholarship.wustl.edu/eng_etds/602

WASHINGTON UNIVERSITY IN ST. LOUIS

McKelvey School of Engineering
Department of Electrical and Systems Engineering

Dissertation Examination Committee:
Joseph A. O'Sullivan, Chair
Mark A. Anastasio, Co-Chair
Ayan Chakrabarti
Abhinav K. Jha
Ulugbek S. Kamilov
Hua Li

Deep Learning for Task-Based Image Quality Assessment in Medical Imaging

by
Weimin Zhou

A dissertation presented to
The Graduate School
of Washington University in
partial fulfillment of the
requirements for the degree
of Doctor of Philosophy

January 2021
St. Louis, Missouri

Table of Contents

List of Figures vi

List of Tables xi

Acknowledgments xiv

Abstract xvii

Chapter 1: Introduction 1

1.1 Overview and Motivation 1

1.2 Outline 6

Chapter 2: Background on statistical inference 8

2.1 Imaging process 8

2.2 Binary signal detection tasks 9

2.2.1 Bayesian Ideal Observer (IO) 10

2.2.2 Hotelling Observer (HO) 13

2.3 Joint signal detection and localization tasks 15

2.3.1 Scanning Ideal Observer 17

2.3.2 Scanning Hotelling Observer 18

Chapter 3: Approximating the Ideal Observer and the Hotelling Observer for binary signal detection tasks by use of supervised learning methods . 20

3.1 Overview 20

3.2 Approximating the IO for signal detection tasks by use of convolutional neural networks (CNNs) 22

3.3 Approximating the HO by use of single layer neural networks 24

3.3.1 Training the HO by use of supervised learning method 24

3.3.2 Learning the HO by use of a Covariance-Matrix Decomposition 26

3.4 Numerical studies 27

3.4.1 Signal-known-exactly (SKE) and background-known-exactly (BKE) signal detection task 28
3.4.2 SKE and background-known-statistically (BKS) signal detection task with a lumpy background model 30
3.4.3 Signal-known-statistically (SKS) and BKS signal detection task with a lumpy background model 31
3.4.4 SKE and BKS signal detection task with a clustered lumpy background model (CLB) 32
3.4.5 Details of training neural networks 35
3.5 Results 38
3.5.1 SKE and BKE signal detection task 38
3.5.2 SKE and BKS signal detection task with a lumpy background 40
3.5.3 SKS and BKS signal detection task with a lumpy background 43
3.5.4 SKE and BKS signal detection task with a CLB 46
3.6 Discussion and Conclusion 51

Chapter 4: Approximating the Ideal Observer for joint signal detection and localization tasks by use of supervised learning method 54
4.1 Overview 54
4.2 Approximating the IO for signal detection-localization tasks by use of CNNs. 56
4.3 Numerical studies 58
4.3.1 BKE signal detection-localization tasks 59
4.3.2 BKS signal detection-localization task with a lumpy background model 61
4.3.3 BKS signal detection-localization task with a CLB model 63
4.3.4 CNN training details 65
4.4 Results 67
4.4.1 BKE signal detection-localization task 67
4.4.2 BKS signal detection-localization task with a lumpy background model 69
4.4.3 BKS signal detection-localization task with a CLB model 70
4.5 Summary 72

Chapter 5: Learning stochastic object models (SOMs) from medical imaging measurements using Progressively-Growing AmbientGANs 74
5.1 Introduction 74

5.2 Background ... 77
5.2.1 Establishing SOMs by use of explicit generative modeling: Propagation of characteristic functionals ... 78
5.2.2 Establishing SOMs by use of implicit generative modeling: Generative adversarial networks (GANs) and AmbientGANs ... 79
5.2.3 Progressively-Growing GAN Training Strategy ... 81
5.3 Establishing SOMs by use of Progressively-Growing AmbientGANs ... 81
5.4 Numerical studies ... 84
5.4.1 Idealized direct imaging system ... 84
5.4.2 Stylized computed tomographic imaging system ... 87
5.4.3 Stylized magnetic resonance (MR) imaging system with complete k-space data ... 89
5.4.4 Stylized MR imaging system with under-sampled k-space data ... 90
5.4.5 Task-based image quality assessment ... 92
5.4.6 Training details ... 93
5.5 Results ... 94
5.5.1 Visual assessments ... 94
5.5.2 Quantitative assessments ... 100
5.5.3 MR imaging system with under-sampled k-space data ... 102
5.5.4 Task-based image quality assessment ... 105
5.6 Discussion and Conclusion ... 106
Chapter 6: Markov-Chain Monte Carlo Approximation of the Ideal Observer using Generative Adversarial Networks ... 108
6.1 Introduction ... 108
6.2 Markov-Chain Monte Carlo method for approximating the IO ... 109
6.3 Markov-Chain Monte Carlo approximation of the IO by use of GANs ... 112
6.4 Numerical studies ... 114
6.4.1 Clinical brain positron emission tomography (PET) images ... 114
6.4.2 Clinical brain MR images ... 117
6.4.3 GAN training details ... 120
6.5 Results ... 122

6.5.1 Clinical brain PET images 122
6.5.2 Clinical brain MR images 124
6.6 Discussion and Conclusion 125
Chapter 7: Summary 128
References 132
Appendix A: Gradient of cross-entropy .... [143]

List of Figures

Figure 3.1: (a)-(c) Samples of the signal-present measurements for the SKE/BKS detection task with the lumpy background model. (d) An image showing the signal contained in (a)-(c). (e)-(g) Samples of the signal-present measurements for the SKE/BKS detection task with the CLB model. (h) An image showing the signal contained in (e)-(g). (© IEEE 2019) 34

Figure 3.2: One instance of the CNN architecture employed for approximating the IO test statistic. (© IEEE 2019) 37

Figure 3.3: Comparison of the Hotelling template in the SKE/BKE case: (a) Analytical Hotelling template; (b) SLNN-HO template; (c) Center line profiles in (a) and (b). The estimated templates are nearly identical. (© IEEE 2019) 38

Figure 3.4: (a) Validation cross-entropy values of CNNs having one to three CONV layers; (b) Testing ROC curves for the IO and HO approximations. (© IEEE 2019) 40

Figure 3.5: Comparison of the Hotelling template in the SKE/BKS case: (a) Traditional Hotelling template; (b) SLNN-HO template; (c) Center line profiles in (a) and (b). The estimated templates are nearly identical. (© IEEE 2019) 41

Figure 3.6: (a) Validation cross-entropy values of CNNs having one to seven CONV layers; (b) Testing ROC curves for the IO and HO approximations. (© IEEE 2019) 42

Figure 3.7: Comparison of the Hotelling template in the SKS/BKS case: (a) Traditional Hotelling template; (b) SLNN-HO template; (c) Center line profiles in (a) and (b). The estimated templates are nearly identical. (© IEEE 2019) 43

Figure 3.8: (a) Validation cross-entropy values produced by CNNs having 1 to 13 CONV layers; (b) Testing ROC curves for the IO and HO approximations. (© IEEE 2019) 45

Figure 3.9: (a) Signal-present measurements; (b) Image showing the signal contained in (a); (c) The signal feature map corresponding to (a); (d) Signal-absent measurements; (e) Image showing that the signal is absent in (d); (f) The signal feature map corresponding to (d). In the signal feature maps, the regions around the signals were activated by the CNN. (© IEEE 2019) 46

Figure 3.10: Comparison of the Hotelling template: (a) Traditional Hotelling template; (b) SLNN-HO template; (c) Center line profiles in (a) and (b). The estimated templates are nearly identical. (© IEEE 2019) 47

Figure 3.11: (a) Validation cross-entropy values of CNNs having one to three CONV layers; (b) Testing ROC curves for the IO and HO approximations. (© IEEE 2019) 48

Figure 3.12: Testing AUC values of CNNs having one to three CONV layers. (© IEEE 2019) 49

Figure 3.13: Curves of validation SNR_t with respect to the number of epochs. (a) Validation SNR_t curve of the SLNN trained on labeled noisy measurements. (b) Validation SNR_t curve of the SLNN trained on background images using decomposition of covariance matrix. The vertical gray line indicates the epoch having the maximum validation SNR_t value. Overfitting occurred after the overall curves of validation SNR_t start to decrease. (© IEEE 2019) 50

Figure 4.1: (a) Signal images corresponding to "System 1". (b) Signal images corresponding to "System 2". (© IEEE 2020) 60

Figure 4.2: (a)-(c) Signal-absent images corresponding to the LB model. (d)-(f) Signal-absent images corresponding to the CLB model. (© IEEE 2020) 64

Figure 4.3: (a) Signal images corresponding to the 9 possible signal locations employed in the BKS task with the LB model. (b) Signal images corresponding to the 9 possible signal locations employed in the BKS task with the CLB model. (© IEEE 2020) 65

Figure 4.4: An instance of the CNN architecture for approximating a set of posterior probabilities for maximizing the ALROC. (© IEEE 2020) ... 67

Figure 4.5: (a) LROC curves corresponding to the IO for the BKE signal detection-localization tasks. (b) ROC curves corresponding to the IO for the simplified binary signal detection tasks. (© IEEE 2020) 68

Figure 4.6: The LROC curves produced by the MCMC-IO (blue), CNN-IO (red-dashed), and the scanning HO (yellow) for the BKS task with the lumpy background model. The LROC curve corresponding to the CNN-IO closely approximates that corresponding to the MCMC-IO and is higher than that produced by the scanning HO. (© IEEE 2020) 70

Figure 4.7: The LROC curves produced by CNN-IO (red) and the scanning HO (yellow) for the BKS task with the CLB model. As expected, the LROC curve corresponding to the CNN-IO is higher than that produced by the scanning HO. (© IEEE 2020) 71

Figure 5.1: An illustration of the AmbientGAN architecture. The generator **G** is trained to generate objects, which are subsequently employed to simulate measurement data. The discriminator **D** is trained to distinguish "real" measurement data to the "fake" measurement data that are simulated by use of the generated objects. 80

Figure 5.2: An illustration of ProAmGAN training. The training starts with low image resolution (e.g., 4×4) and the image resolution is increased progressively by adding more layers to the generator and the discriminator. The discriminator is trained to distinguish between the ground-truth and generated reconstructed objects........ 83

Figure 5.3: An illustration of idealized planar X-ray imaging system that acquires noisy imaging measurements. 85

Figure 5.4: An illustration of tomographic imaging system that acquires Radon transform data. 88

Figure 5.5: MR imaging system with complete k-space data. Logarithm of one plus the magnitude of k-space data was displayed. 89

Figure 5.6: Top: k-space sampling patterns corresponding to different sampling ratios of 1/1, 4/5, 1/2, 1/4, and 1/8 from left to right; Bottom: images reconstructed by use of $\mathbf{H}^{\dagger}$ corresponding to the k-space sampling patterns in the top row. 91

Figure 5.7: Five signals considered in the signal detection study. 93

Figure 5.8: Top: Ground-truth chest X-ray objects $\mathbf{f}$. Bottom: ProAmGAN-generated chest X-ray objects $\hat{\mathbf{f}}$. 95

Figure 5.9: ProAmGAN-generated chest X-ray images at different training steps. FID scores decreased as the resolution increased in the training process. 95

Figure 5.10: A ProGAN-generated (left panel) and ProAmGAN-generated (right panel) chest X-ray object........ 96

Figure 5.11: Top: Ground-truth chest CT objects $\mathbf{f}$. Bottom: ProAmGAN-generated chest CT objects $\hat{\mathbf{f}}$. 97

Figure 5.12: ProAmGAN-generated chest CT images at different training steps. FID scores decreased as the resolution increased in the training process. 97

Figure 5.13: A ProGAN-generated (left panel) and ProAmGAN-generated (right panel) chest CT object. 98

Figure 5.14: Top: Ground-truth brain MR objects $\mathbf{f}$. Bottom: ProAmGAN-generated brain MR objects $\hat{\mathbf{f}}$. 99

Figure 5.15: ProAmGAN-generated brain MR images at different training steps. FID scores decreased as the resolution increased in the training process. 99

Figure 5.16: A ProGAN-generated (left panel) and ProAmGAN-generated (right panel) brain MR object. 100

Figure 5.17: Empirical PDFs of SSIMs corresponding to ground-truth image pairs (red curves), ground-truth and ProAmGAN-generated image pairs (blue curves), and ground-truth and ProGAN-generated image pairs (yellow curves). 102

Figure 5.18: Top: Examples of ground-truth objects $\mathbf{f}$. Bottom: Examples of ProAmGAN-generated objects $\hat{\mathbf{f}}$ corresponding to the data-acquisition design with 4/5 k-space sampling ratio. 103

Figure 5.19: ProAmGAN-generated objects (top row) and ProGAN-generated objects (bottom row). From left to right, the ProGAN and ProAmGAN trained with the k-space sampling ratio of 1/1, 4/5, 1/2, 1/4, and 1/8. 104

Figure 5.20: Hotelling observer performance corresponding to different tasks with different signals, noise levels, and k-space sampling ratios. 105

Figure 6.1: (a) An image from the ADNI PET dataset. (b) A simulated low-dose PET image corresponding to (a). (c) The signal image corresponds to the considered signal detection task. 115

Figure 6.2: (a) An image from the ADNI MR dataset. (b) A signal-absent image that is generated by adding i.i.d. Gaussian noise to (a). (c) The signal image corresponds to the considered signal detection task. 118

Figure 6.3: Top: Examples of ground-truth PET images. Bottom: Examples of ProGAN-generated PET images. 122

Figure 6.4: The ROC curve corresponding to the MCMC-GAN IO, CNN-IO and the HO. The ROC curve corresponding to the MCMC-GAN IO is in close agreement with the CNN-IO and is higher than the HO. 123

Figure 6.5: Top: Examples of ground-truth MR images. Bottom: Examples of ProGAN-generated MR images. 124

Figure 6.6: The ROC curve corresponding to the MCMC-GAN IO, CNN-IO and the HO. The ROC curve corresponding to the MCMC-GAN IO is in close agreement with the CNN-IO and is higher than the HO. 125

List of Tables

Table 3.1: Parameters for generating CLB images. 33

Table 3.2: SNR^2_{HO} computed from both background images $\mathbf{b}$ and measurements $\mathbf{g}$. The Hotelling template computed from few images can cause significant positive bias. However, when SLNNs were trained using our proposed methods, early-stopping strategy in which the epoch having the maximum validation SNR_t was selected could be employed to closely approximate the HO performance. (© IEEE 2019) 51

Table 4.1: Parameters for generating CLB images 64

Table 5.1: The architectures of the generator and discriminator for generating 512 × 512 images (a) and those for generating 256 × 256 images (b). More details about each component in the architecture can be found in ProGAN paper [59] 86

Table 5.2: FID and metrics that evaluate PDFs of SSIMs. Here, “X-ray”, “CT”, and “MRI” correspond to the idealized direct imaging system, computed tomographic imaging system and MR imaging system with complete k-space data, respectively. 101

Table 5.3: FID scores corresponding to the objects and the measurement components. 104

Table 6.1: The architecture of the generator and discriminator for establishing SOMs corresponding to the considered brain PET and MR images. More details about each component in the architecture can be found in ProGAN paper [59] 121

List of Abbreviations

ANN Artificial Neural Network

BKE Background-known-exactly

BKS Background-known-statistically

CF Characteristic Function

CFl Characteristic Functional

CLB Clustered Lumpy Background

CNN Convolutional Neural Network

CT Computed Tomography

FCNN Fully-connected Neural Network

FID Fréchet Inception Distance

FOM Figure-of-merit

FPF False-positive Fraction

GAN Generative Adversarial Network

HMC Hamiltonian Monte Carlo

HO Hotelling Observer

IO Ideal Observer

IQ Image Quality

LROC Localization Receiver Operating Characteristic

MALA Metropolis Adjusted Langevin Algorithm

MCMC Markov-Chain Monte Carlo

MGLRT Modified Generalized Likelihood Ratio Test

ML Maximum Likelihood

MR Magnetic Resonance

MSE Mean Squared Error

PDF Probability Density Function

PET Positron Emission Tomography

PRF Point Response Function

ROC Receiver Operating Characteristic

RWMH Random Walk Metropolis-Hastings

SLNN Single-layer Neural Network

SKE Signal-known-exactly

SKS Signal-known-statistically

SOM Stochastic Object Model

SSIM Structural Similarity Index

TPF True-positive Fraction

Acknowledgments

I would like to express my most sincere thankfulness to my advisor, Dr. Mark A. Anastasio, for his insightful guidance, excellent mentorship, and kind support during these past few years. It has been my great pleasure to learn from him. I have been benefited from not only his profound knowledge, but also his easy-going personality as a great friend.

I am grateful to my other committee members, Dr. Joseph A. O'Sullivan, Dr. Ulugbek S. Kamilov, Dr. Abhinav K. Jha, Dr. Ayan Chakrabarti, and Dr. Hua Li, for their valuable suggestions and support.

I would like to thank all of my current and previous labmates in the Computational Imaging Science Laboratory for the helpful discussions during these years. Particularly, I would like to thank Dr. Yang Lou, Dr. Yujia Chen, Dr. Huifeng Guan, Dr. Trey Garson, Dr. Thomas P. Matthews, and Dr. Joemini Poudel for their insightful discussions and patient guidance. I would also like to thank Mr. Sayantan Bhadra, Mr. Shenghua He, and Mr. Kaiyan Li for their excellent work during our collaborations.

Last but not the least, I would like to convey my special thanks to my parents and my wife for their love, caring, and encouragement. Their unconditional love has made this achievement possible.

Weimin Zhou

Washington University in Saint Louis
January 2021

Dedicated to my parents.

ABSTRACT OF THE DISSERTATION

Deep Learning for Task-Based Image Quality Assessment in Medical Imaging

by

Weimin Zhou

Doctor of Philosophy in Electrical Engineering

Washington University in St. Louis, 2021

Professor Joseph A. O'Sullivan, Chair

Professor Mark A. Anastasio, Co-Chair

It has been advocated to use objective measures of image quality (IQ) for assessing and optimizing medical imaging systems. Objective measures of IQ quantify the performance of an observer at a specific diagnostic task. Binary signal detection tasks and joint signal detection and localization (detection-localization) tasks are commonly considered in medical imaging. When optimizing imaging systems for binary signal detection tasks, the performance of the Bayesian Ideal Observer (IO) has been advocated for use as a figure-of-merit (FOM). The IO maximizes the observer performance that is summarized by the receiver operating characteristic (ROC) curve. When signal detection-localization tasks are considered, the IO that implements a modified generalized likelihood ratio test (MGLRT) maximizes the observer performance as measured by the localization ROC (LROC) curve. However, computation of the IO test statistic generally is analytically intractable. To address this difficulty, sampling-based methods that employ Markov-Chain Monte Carlo (MCMC) techniques have been proposed. However, current applications of MCMC methods have been limited to relatively simple stochastic object models (SOMs). When the IO is difficult or intractable to compute, the optimal linear observer, known as the Hotelling Observer (HO), can be employed to evaluate objective measures of IQ. Although computation of the HO is easier than that of the

IO, it can still be challenging or even intractable because a potentially large covariance matrix needs to be estimated and subsequently inverted. In the first part of the dissertation, we introduce supervised learning-based methods for approximating the IO and the HO for binary signal detection tasks. The use of convolutional neural networks (CNNs) to approximate the IO and the use of single layer neural networks (SLNNs) to directly estimate the Hotelling template without computing and inverting covariance matrices are demonstrated. In the second part, a supervised learning method that employs CNNs to approximate the IO for signal detection-localization tasks is presented. This method represents a deep-learning-based implementation of a MGLRT that defines the IO decision strategy for signal detection-localization tasks.

When evaluating observer performance for assessing and optimizing imaging systems by use of objective measures of IQ, all sources of variability in the measured image data should be accounted for. One important source of variability that can significantly affect observer performance is the variation in the ensemble of objects to-be-imaged. To describe this variability, a SOM can be established. A SOM is a generative model that can produce an ensemble of simulated objects with prescribed statistical properties. In order to establish a realistic SOM, it is desirable to use experimental data. Generative adversarial networks (GANs) hold great potential for establishing SOMs. However, images produced by imaging systems are affected by the measurement noise and a potential reconstruction process. Therefore, GANs that are trained by use of these images cannot represent SOMs because they are not established to learn object variability alone. An augmented GAN architecture named AmbientGAN that includes a measurement operator was proposed to address this issue. However, AmbientGANs cannot be immediately implemented with advanced GAN training strategies such as progressive growing of GANs (ProGANs). Therefore, the ability of AmbientGANs to establish realistic and sophisticated SOMs is limited. In the third part

of this dissertation, we propose a novel deep learning method named progressively growing AmbientGANs (ProAmGANs) that incorporates the advanced progressive growing training procedure and therefore enables the AmbientGAN to be applied to realistically sized medical image data. Stylized numerical studies involving a variety of object ensembles with common medical imaging modalities are presented.

Finally, a novel sampling-based method named MCMC-GAN is developed to approximate the IO. This method applies MCMC algorithms to SOMs that are established by use of GAN techniques. Because the implementation of GANs is general and not limited to specific images, our proposed method can be implemented with sophisticated object models and therefore extends the domain of applicability of the MCMC techniques. Numerical studies involving clinical brain positron emission tomography (PET) images and brain magnetic resonance (MR) images are presented.

Chapter 1

Introduction

The goals of this dissertation are to investigate and develop deep learning approaches to facilitate the assessment and optimization of imaging systems and data-acquisition designs by use of objective measures of image quality (IQ). In this chapter, a brief overview of objective measures of IQ and motivations of the dissertation are presented.

1.1 Overview and Motivation

Medical imaging systems and data-acquisition designs are commonly assessed and optimized by use of objective measures of image quality (IQ) that quantify the performance of an observer at specific tasks [9, 68, 81, 82, 96, 118, 119, 123, 125]. Common diagnostic tasks in medical imaging include signal detection tasks (e.g., detection of a tumor) and signal detection-localization tasks (e.g., joint detection and localization of a tumor). The receiver operating characteristic (ROC) curve is commonly employed to summarize observer performance on binary signal detection tasks [9]. When optimizing imaging systems for binary signal detection tasks, the performance of the Bayesian Ideal Observer (IO) has been advocated for use in computing a figure-of-merit (FOM). In this way, imaging systems can be optimized so that the amount of task-specific information in the measured image data is maximized. The IO implements a test statistic that can be any monotonic transformation of the likelihood ratio and maximizes the observer performance as measured by the area under the ROC curve (AUC) [9]. When signal detection-localization tasks are considered, the localization ROC (LROC) curve can be employed to summarize observer performance, and the IO that implements a modified generalized likelihood ratio test (MGLRT) maximizes the observer performance as measured by the area under the LROC curve (ALROC) [62]. The IO performance measured by the ALROC can be used to provide a FOM for assessing and optimizing imaging systems for signal detection-localization tasks. The IO performance can also be employed to assess the efficiency of the human observer and other model observers [18].

However, computation of the IO test statistic is analytically intractable in the majority of cases. Sampling-based methods that employ Markov-Chain Monte Carlo (MCMC) techniques have been developed to numerically estimate likelihood ratios [68]. However, to properly implement MCMC methods, many practical issues such as the design of proposal density function from which the Markov chain can be efficiently generated need to be addressed. Current applications of MCMC methods have been limited to relatively simple stochastic object models (SOMs) such as a lumpy object model [68, 85], a binary texture model [2] and a parametrized torso phantom [50]. The applicability of the MCMC methods to other more sophisticated SOMs remains under-explored. Other strategies that employ the Fisher information with surrogate FOMs have been developed for circumventing the difficulty of the IO computation [23, 27, 56].

When the IO test statistic is difficult to compute, the optimal linear observer, known as the Hotelling Observer (HO), can be employed to assess and optimize imaging systems and data-acquisition designs [9]. The HO is optimal in the sense that it maximizes the signal-to-noise ratio (SNR) of the test statistic among all linear observers [9]. Although the HO computation is typically easier than the IO computation, it is not without any challenge. Specifically, the implementation of the HO requires the estimation and inversion of a covariance matrix, which can be computational challenging and even intractable when high-dimensional image data are considered. To circumvent this limitation, dimensionality reduction methods that employ channel mechanisms have been developed [9, 16, 38, 69, 81]. However, to properly implement such methods for approximating the HO, the design of efficient channels that can preserve task-specific information in the channelized data needs to be addressed.

Supervised learning methods hold significant promise for designing and implementing model observers [3, 16, 75, 107]. Imaging systems and data-acquisition designs are commonly assessed and optimized via computer-simulation. In such applications, large amounts of data can be simulated for training complicated inference models to be used for assessing objective measures of IQ. It is known that artificial neural networks (ANNs) that possess sufficient computational capacities can approximate any continuous function [54]. Therefore, in principle, ANNs can be trained to approximate functions that compute test statistics of model observers. A previous work demonstrated the ability of fully-connected neural networks (FCNNs) to approximate the IO acting on low-dimensional feature vectors that are extracted from image data [68]. However, because each neuron in FCNNs is fully connected to all neurons in the previous layer, the dimension of the input layer and the depth of the network that can be trained effectively are limited. As such, FCNNs do not scale well to large dimensional data and are not well suited to be employed for approximating the IO that directly acts on image data.

Convolutional neural networks (CNNs) have been developed to circumvent this limitation, and they have achieved success in many image classification tasks [22, 40, 70, 71]. In Chapter 3, supervised learning methods that employ CNNs to approximate the IO that directly acts on image data for binary signal detection tasks are introduced. Supervised learning methods that employ single-layer neural networks (SLNNs) to approximate the HO are also described. This method enables a direct estimation of the Hotelling template without explicitly estimating and inverting covariance matrices. In Chapter 4, a supervised learning method that employs CNNs to approximate the IO for signal detection-localization tasks are presented. This method represents a deep-learning-based implementation of the IO decision strategy defined by a MGLRT.

When the performance of an observer at specific tasks is evaluated for assessing and optimizing imaging systems, all sources of variability in the measured image data should be accounted for. The variation in the ensemble of objects to-be-imaged is an important source of variability that can significantly limit observer performance. This variability can be described by a stochastic object model (SOM). A SOM is a generative model that can produce an ensemble of objects with prescribed statistical properties. Previously established SOMs include lumpy object models [91] , clustered lumpy object models [13], and binary texture models [2]. Other mathematical or voxelized computational phantoms [19, 28, 72, 94, 95, 109, 112, 126] have been proposed for simulating medical images. However, the majority of these phantoms were established by use of only few subjects. Therefore, they may not be able to completely and accurately describe the statistical properties of the ensemble of objects. A variety of anatomical shape models have also been developed to describe common geometric features and geometric variability for shape analysis applications [4, 29, 30, 37, 47, 51, 97, 103]. However, to date, these models have not been systematically explored for the purpose of establishing SOMs that capture realistic object textures and anatomical variations for use in assessing objective measures of IQ.

In order to establish SOMs that capture realistic object variability, it is desirable to use experimental data. However, SOMs that are defined to provide an *in silico* representation of the ensemble of objects to-be-imaged should be independent of the imaging system, measurement noise and reconstruction algorithm. To address this need, Kupinski *et al.* proposed an explicit generative modeling method to estimate parameters associated with some specific SOMs by use of noisy imaging measurement data and well-characterized imaging system models [66]. Implementations of this method to establish lumpy and clustered lumpy object models have been demonstrated [66]. However, this method has been limited to situations where the characteristic function of the imaging measurement data can be analytically computed. There is still a significant need to develop a generalized method to establish more realistic and complicated SOMs.

Deep learning methods that employ generative adversarial networks (GANs) [46] hold great potential to establish SOMs that can produce an ensemble of objects having statistical properties that are consistent with training data. However, conventional GANs are typically trained with reconstructed images that are influenced by the measurement noise and reconstruction process. To address this issue, an augmented GAN architecture named AmbientGAN [15] has been proposed. The AmbientGAN architecture augments the conventional GAN architecture with a measurement process. This architecture enables the establishment of generative models that describe object variability from noisy and indirect imaging measurement data. However, similar to conventional GANs, the training process of AmbientGANs can be unstable. This training instability limits the ability of AmbientGANs to generate high-dimensional medical images that depict object properties of interest.

An advanced GAN training strategy named progressive growing of GANs (ProGANs) [59] has been introduced to improve the training stability of GANs. ProGANs adopt a multi-resolution strategy to stabilize the adversarial training process. It has been demonstrated that ProGANs can be successfully employed to generate high-resolution images [59]. However, similar to conventional GANs, ProGANs that are trained on images produced by imaging systems can not represent SOMs because they are affected by measurement noise and a potential reconstruction operator. In Chapter 5, a novel AmbientGAN method named progressively growing AmbientGANs (ProAmGANs) is proposed that permits the implementation of the progressive growing strategy with the training of AmbientGANs. The proposed ProAmGAN method enables the establishment of SOMs from imaging measurements that can yield high-dimensional images.

The ability of modern GAN techniques to establish realistic SOMs also enables the development of sampling-based methods for approximating the IO for realistic diagnostic tasks. In Chapter 6, a novel sampling-based method named MCMC-GAN for approximating the IO is provided. This method approximates the IO test statistic by applying MCMC techniques with SOMs learned by use of GANs. Compared to the previous MCMC methods that have been limited to some relatively simple SOMs, MCMC-GAN can be implemented with more realistic and sophisticated SOMs that can be learned by use of GANs. Therefore, this method extends the domain of applicability of MCMC approaches to approximate the IO for assessing and optimizing imaging systems and data-acquisition designs.

1.2 Outline

The dissertation provides background information on objective-measures of IQ in Chapter 2. Signal detection theory and previous works on approximating the IO and HO are reviewed.

In Chapter 3, a supervised learning method that employs CNNs to approximate the IO acting on image data for binary signal detection tasks is introduced. Supervised learning-based methods that employ SLNNs to approximate the HO are also provided. This work was proposed and described previously [118, 123, 125].

In Chapter 4, a supervised learning-based method that employs CNNs to approximate the IO for signal detection-localization tasks is discussed. This work was proposed and described previously [117, 124].

In Chapter 5, a novel AmbientGAN training approach named progressively growing AmbientGANs (ProAmGANs) to learn realistic SOMs from imaging measurements is presented. This work was proposed and described previously [120, 121].

In Chapter 6, a novel sampling-based method named MCMC-GAN to approximate the IO is provided. This method extends the domain of applicability of MCMC techniques to approximate the IO for assessing objective measures of IQ. A preliminary study of this method was described previously [119].

Finally, the dissertation is summarized with a discussion in Chapter 7.

Chapter 2

Background on statistical inference

Statistical inference can be divided into two categories: classification and parameter estimation [9]. This chapter provides background information on classification tasks that are often considered for evaluating objective measures of image quality in medical imaging.

2.1 Imaging process

A digital imaging system can be mathematically described as:

$$\mathbf{g} = \mathcal{H} f(\mathbf{r}) + \mathbf{n}, \tag{2.1}$$

where $\mathbf{g} \in \mathbb{R}^M$ denotes the measured image data, $f(\mathbf{r})$ is a function of a spatial coordinate $\mathbf{r} \in \mathbb{R}^d$ that describes the object being imaged, $\mathcal{H}$ denotes a continuous-to-discrete (C-D) imaging operator that maps $\mathbb{L}_2(\mathbb{R}^d) \rightarrow \mathbb{R}^M$, and $\mathbf{n} \in \mathbb{R}^M$ is the measurement noise. Because the measurement noise $\mathbf{n}$ is random, the measured image data $\mathbf{g}$ is random. Object variability is known to limit observer performance [84]. Therefore, the object function $f(\mathbf{r})$ can be either deterministic or stochastic, depending on the specification of the diagnostic task to be assessed. When a linear imaging operator is considered, the measured image data can be described as:

$$g_m = \int_{\mathbb{R}^d} d\mathbf{r}\ h_m(\mathbf{r}) f(\mathbf{r}) + n_m, \tag{2.2}$$

where g_m is the m^{th} element of the vector $\mathbf{g}$, n_m is the m^{th} element of the vector $\mathbf{n}$, and $h_m(\mathbf{r})$ is the point response function (PRF) that describes the sensitivity of the m^{th} measurement g_m to the object function $f(\mathbf{r})$ at point $\mathbf{r}$. The notation $\mathbf{f}$ will be employed to denote $f(\mathbf{r})$ when the spatial dependence of the object function is not important to highlight.

Below, reviews of binary signal detection tasks and signal detection-localization tasks that are frequently considered in medical imaging are provided.

2.2 Binary signal detection tasks

When a binary signal detection task is considered, an observer is required to classify an image as satisfying either a signal-absent hypothesis (H_0) or a signal-present hypothesis (H_1). The imaging processes under H_0 and H_1 can be described as:

$$\begin{aligned} H_0 &: \mathbf{g} = \mathbf{b} + \mathbf{n}, \\ H_j &: \mathbf{g} = \mathbf{b} + \mathbf{s} + \mathbf{n}, \end{aligned} \tag{2.3}$$

where $\mathbf{b} \equiv \mathcal{H} f_b(\mathbf{r})$ is the image of the background $f_b(\mathbf{r})$, and $\mathbf{s} \equiv \mathcal{H} f_s(\mathbf{r})$ is the image of the signal $f_s(\mathbf{r})$. Denote the m^{th} component of $\mathbf{b}$ and $\mathbf{s}$ as b_m and s_m, respectively. When the imaging operator $\mathcal{H}$ is linear, these quantities can be computed as:

$$\begin{aligned} b_m &= \int_{\mathbb{R}^d} d\mathbf{r}\ h_m(\mathbf{r}) f_b(\mathbf{r}), \\ s_m &= \int_{\mathbb{R}^d} d\mathbf{r}\ h_m(\mathbf{r}) f_s(\mathbf{r}). \end{aligned} \tag{2.4}$$

When background-known-exactly (BKE) signal detection tasks are considered, $f_b(\mathbf{r})$ is deterministic, whereas when background-known-statistically (BKS) signal detection tasks are considered, $f_b(\mathbf{r})$ is random. Similarly, when signal-known-exactly (SKE) signal detection tasks are considered, $f_s(\mathbf{r})$ is deterministic, whereas when signal-known-statistically (SKS) signal detection tasks are considered, $f_s(\mathbf{r})$ is random.

To perform a binary signal detection task, an observer computes a test statistic $t(\mathbf{g})$ that maps the measured image data $\mathbf{g}$ to a real-valued scalar. The test statistic $t(\mathbf{g})$ is compared to a pre-determined threshold τ to classify $\mathbf{g}$: if $t(\mathbf{g}) > \tau$, classify $\mathbf{g}$ as satisfying H_1, otherwise, classify $\mathbf{g}$ as satisfying H_0. A receiver operating characteristic (ROC) curve that depicts the trade-off between the false-positive fraction (FPF) and the true-positive fraction (TPF) can be plotted by varying the threshold τ. The area under the ROC curve (AUC) can be subsequently computed to quantify the observer performance.

2.2.1 Bayesian Ideal Observer (IO)

The Bayesian Ideal Observer (IO) employs complete statistical knowledge and sets an upper performance limit among all observers. The IO computes a test statistic that can be any monotonic transformation of the likelihood ratio $\Lambda(\mathbf{g})$, which is defined as [9, 67, 68]:

$$\Lambda(\mathbf{g}) = \frac{p(\mathbf{g}|H_1)}{p(\mathbf{g}|H_0)}, \tag{2.5}$$

where $p(\mathbf{g}|H_j)$ is the conditional probability density function that describes the likelihood of the measured image data $\mathbf{g}$ under the hypothesis H_j $(j = 0, 1)$. One important monotonic transformation of $\Lambda(\mathbf{g})$ that will prove useful is the posterior probability $\Pr(H_1|\mathbf{g})$:

$$\Pr(H_1|\mathbf{g}) = \frac{[\Pr(H_1)/\Pr(H_0)]\Lambda(\mathbf{g})}{1 + [\Pr(H_1)/\Pr(H_0)]\Lambda(\mathbf{g})}, \tag{2.6}$$

where $\Pr(H_0)$ and $\Pr(H_1)$ are the prior probabilities associated with the two hypotheses.

However, computation of the IO test statistic is analytically intractable in the majority of cases. To address this issue, Kupinski *et al.* proposed a sampling based method that employs Markov-Chain Monte Carlo (MCMC) techniques to numerically compute the likelihood ratio $\Lambda(\mathbf{g})$ [68]. When a signal-known-exactly (SKE) binary signal detection task is considered, the likelihood ratio $\Lambda(\mathbf{g})$ can be computed as [68]:

$$\Lambda(\mathbf{g}) = \frac{\int d\mathbf{b}\ p_b(\mathbf{b})p(\mathbf{g}|\mathbf{b}, H_1)}{\int d\mathbf{b}\ p_b(\mathbf{b})p(\mathbf{g}|\mathbf{b}, H_0)} \equiv \int d\mathbf{b}\ \Lambda_{\mathrm{BKE}}(\mathbf{g}|\mathbf{b})p(\mathbf{b}|\mathbf{g}, H_0), \tag{2.7}$$

where $\Lambda_{\mathrm{BKE}}(\mathbf{g}|\mathbf{b})$ is the likelihood ratio given a background $\mathbf{b}$ and $p(\mathbf{b}|\mathbf{g}, H_0)$ is a posterior probability density function. These quantities can be computed as:

$$\Lambda_{\mathrm{BKE}}(\mathbf{g}|\mathbf{b}) = \frac{p(\mathbf{g}|\mathbf{b}, H_1)}{p(\mathbf{g}|\mathbf{b}, H_0)}. \tag{2.8a}$$

$$p(\mathbf{b}|\mathbf{g}, H_0) = \frac{p(\mathbf{g}|\mathbf{b}, H_0)p_b(\mathbf{b})}{\int d\mathbf{b}' p(\mathbf{g}|\mathbf{b}', H_0)p_b(\mathbf{b}')}. \tag{2.8b}$$

When the background $\mathbf{b}$ can be described by a stochastic object model (SOM) that takes a set of random variables $\boldsymbol{\theta}$ as the input, i.e., $\mathbf{b} \equiv \mathbf{b}(\boldsymbol{\theta})$, the likelihood ratio described in Eq. (2.7) can be computed as [68]:

$$\Lambda(\mathbf{g}) = \int d\boldsymbol{\theta}\ \Lambda_{\mathrm{BKE}}(\mathbf{g}|\mathbf{b}(\boldsymbol{\theta}))p(\boldsymbol{\theta}|\mathbf{g}, H_0). \tag{2.9}$$

Monte Carlo integration can be subsequently employed to approximate the likelihood ratio [68]:

$$\hat{\Lambda}(\mathbf{g}) = \frac{1}{N_c}\sum_{i=1}^{N_c} \Lambda_{\mathrm{BKE}}(\mathbf{g}|\mathbf{b}(\boldsymbol{\theta}^i)), \tag{2.10}$$

where $\hat{\Lambda}(\mathbf{g})$ is the MCMC approximation of the likelihood ratio $\Lambda(\mathbf{g})$, N_c is the number of samples used in Monte Carlo integration, and $\boldsymbol{\theta}^i$ is drawn from the posterior probability function $p(\boldsymbol{\theta}|\mathbf{g}, H_0)$. To draw samples $\boldsymbol{\theta}^i$ from the posterior probability function $p(\boldsymbol{\theta}|\mathbf{g}, H_0)$, a Markov Chain having the stationary density $p(\boldsymbol{\theta}|\mathbf{g}, H_0)$ can be generated. To achieve this, a Metropolis-Hastings algorithm can be implemented. Specifically, an initial vector $\boldsymbol{\theta}^0$ is selected and a proposal density function $q(\boldsymbol{\theta}|\boldsymbol{\theta}^i)$ is specified. For a vector $\boldsymbol{\theta}^i$, a candidate vector $\hat{\boldsymbol{\theta}}$ is sampled from the proposal density $q(\boldsymbol{\theta}|\boldsymbol{\theta}^i)$ and is accepted with the probability $p_{\mathrm{a}}(\widehat{\boldsymbol{\theta}}|\boldsymbol{\theta}^j, \mathbf{g})$, which is defined as:

$$p_{\mathrm{a}}(\hat{\boldsymbol{\theta}}|\boldsymbol{\theta}^j, \mathbf{g}) = \min\left[1, \frac{p(\mathbf{g}|\mathbf{b}(\hat{\boldsymbol{\theta}}), H_0)p(\hat{\boldsymbol{\theta}})q(\boldsymbol{\theta}^j|\hat{\boldsymbol{\theta}})}{p(\mathbf{g}|\mathbf{b}(\boldsymbol{\theta}^j), H_0)p(\boldsymbol{\theta}^j)q(\hat{\boldsymbol{\theta}}|\boldsymbol{\theta}^j)}\right]. \tag{2.11}$$

If the candidate vector $\hat{\boldsymbol{\theta}}$ is accepted, it is added to the Markov Chain: $\boldsymbol{\theta}^{i+1} = \hat{\boldsymbol{\theta}}$; otherwise, $\boldsymbol{\theta}^{i+1} = \boldsymbol{\theta}^i$.

Park *et al.* extended this MCMC approach to signal-known-statistically (SKS) signal detection tasks [85]. If the signal can be described by a stochastic model that takes a set of random variables $\boldsymbol{\alpha}$ as the input, i.e., $\mathbf{s} = \mathbf{s}(\boldsymbol{\alpha})$, the likelihood ratio $\Lambda(\mathbf{g})$ can be computed as [85]:

$$\Lambda(\mathbf{g}) = \int d\boldsymbol{\alpha} \int d\boldsymbol{\theta}\; \Lambda_{\mathrm{BSKE}}(\mathbf{g}|\mathbf{b}(\boldsymbol{\theta}), \mathbf{s}(\boldsymbol{\alpha}))p(\boldsymbol{\theta}|\mathbf{g}, H_0)p(\boldsymbol{\alpha}), \tag{2.12}$$

where $\Lambda_{\mathrm{BSKE}}(\mathbf{g}|\mathbf{b}(\boldsymbol{\theta}), \mathbf{s}(\boldsymbol{\alpha}))$ is the likelihood ratio given a background $\mathbf{b}(\boldsymbol{\theta})$ and a signal $\mathbf{s}(\boldsymbol{\alpha})$:

$$\Lambda_{\mathrm{BSKE}}(\mathbf{g}|\mathbf{b}(\boldsymbol{\theta}), \mathbf{s}(\boldsymbol{\alpha})) = \frac{p(\mathbf{g}|\mathbf{b}(\boldsymbol{\theta}), \mathbf{s}(\boldsymbol{\alpha}), H_1)}{p(\mathbf{g}|\mathbf{b}(\boldsymbol{\theta}), H_0)}. \tag{2.13}$$

The likelihood ratio $\Lambda(\mathbf{g})$ can be subsequently approximated as:

$$\hat{\Lambda}(\mathbf{g}) = \frac{1}{J}\sum_{j=1}^{J} \Lambda_{\mathrm{BSKE}}(\mathbf{g}|\mathbf{b}(\boldsymbol{\theta}^j), \mathbf{s}(\boldsymbol{\alpha}^j)). \tag{2.14}$$

Here, $(\boldsymbol{\theta}^j, \boldsymbol{\alpha}^j)$ are sampled from the probability density function $p(\boldsymbol{\theta}|\mathbf{g}, H_0)p(\boldsymbol{\alpha})$ by use of MCMC methods. Again, to construct a Markov chain, a Metropolis-Hastings algorithm with a proposal density $q(\boldsymbol{\theta}|\boldsymbol{\theta}^i)$ and $q(\boldsymbol{\alpha}|\boldsymbol{\alpha}^i)$ can be employed. The acceptance probability of a candidate vector $(\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{\alpha}})$ can be computed as:

$$p_{\mathrm{a}}(\widehat{\boldsymbol{\theta}}, \widehat{\boldsymbol{\alpha}}|\boldsymbol{\theta}^j, \boldsymbol{\alpha}^j, \mathbf{g}) = \min\left[1, \frac{p(\mathbf{g}|\mathbf{b}(\hat{\boldsymbol{\theta}}), H_0)p(\hat{\boldsymbol{\theta}})p(\hat{\boldsymbol{\alpha}})q(\boldsymbol{\theta}^j|\hat{\boldsymbol{\theta}})q(\boldsymbol{\alpha}^j|\hat{\boldsymbol{\alpha}})}{p(\mathbf{g}|\mathbf{b}(\boldsymbol{\theta}^j), H_0)p(\boldsymbol{\theta}^j)p(\boldsymbol{\alpha}^j)q(\hat{\boldsymbol{\theta}}|\boldsymbol{\theta}^j)q(\hat{\boldsymbol{\alpha}}|\boldsymbol{\alpha}^j)}\right]. \tag{2.15}$$

However, current applications of MCMC methods have been limited to relatively simple SOMs such as a lumpy object model [68], a binary texture model [2], and a parameterized torso phantom [50]. It remains unclear how to implement these MCMC methods with other SOMs.

2.2.2 Hotelling Observer (HO)

When the IO test statistic is intractable or difficult to compute, the Hotelling observer (HO) can be employed to assess objective measures of IQ for optimizing imaging systems and data-acquisition designs. The HO test statistic $t_{\mathrm{HO}}(\mathbf{g})$ can be computed as [9]:

$$t_{\mathrm{HO}}(\mathbf{g}) = \mathbf{w}_{\mathrm{HO}}^T\mathbf{g}, \tag{2.16}$$

where $\mathbf{w}_{\text{HO}} \in \mathbb{R}^{M\times 1}$ is the Hotelling template, which is defined as [9]:

$$\mathbf{w}_{\text{HO}} = \left[\frac{1}{2}(\mathbf{K}_0 + \mathbf{K}_1)\right]^{-1} \Delta\bar{\bar{\mathbf{g}}}. \tag{2.17}$$

Here, $\mathbf{K}_j$ is the covariance matrix of the measured image data $\mathbf{g}$ under the hypothesis H_j $(j = 0, 1)$ and $\Delta\bar{\bar{\mathbf{g}}}$ is the difference between the mean of the measured image data $\mathbf{g}$ under the hypothesis H_1 and H_0. Denote the conditional mean of the measured image data $\mathbf{g}$ given an object $\mathbf{f}$ as $\bar{\mathbf{g}}(\mathbf{f}) \equiv \langle \mathbf{g} \rangle_{\mathbf{g}|\mathbf{f}}$, and denote the mean of $\bar{\mathbf{g}}(\mathbf{f})$ under the hypothesis H_j as $\bar{\bar{\mathbf{g}}}_j \equiv \langle \bar{\mathbf{g}}(\mathbf{f}) \rangle_{\mathbf{f}|H_j}$. The quantities $\Delta\bar{\bar{\mathbf{g}}}$ and $\mathbf{K}_j$ can computed as:

$$\Delta\bar{\bar{\mathbf{g}}} = \bar{\bar{\mathbf{g}}}_1 - \bar{\bar{\mathbf{g}}}_0, \tag{2.18a}$$

$$\mathbf{K}_j = \left\langle \langle [\mathbf{g} - \bar{\bar{\mathbf{g}}}_j][\mathbf{g} - \bar{\bar{\mathbf{g}}}_j]^T \rangle_{\mathbf{g}|\mathbf{f}} \right\rangle_{\mathbf{f}|H_j}. \tag{2.18b}$$

When computing the Hotelling template $\mathbf{w}_{\text{HO}}$, it is sometimes useful to decompose the covariance matrix $\mathbf{K}_j$ as [9]:

$$\mathbf{K}_j = \langle \mathbf{K}_{\mathbf{n}|\mathbf{f}} \rangle_{\mathbf{f}|H_j} + \mathbf{K}_{\bar{\mathbf{g}}(\mathbf{f})|H_j}, \tag{2.19}$$

where $\mathbf{K}_{\mathbf{n}|\mathbf{f}} = \langle [\mathbf{g} - \bar{\mathbf{g}}(\mathbf{f})][\mathbf{g} - \bar{\mathbf{g}}(\mathbf{f})]^T \rangle_{\mathbf{g}|\mathbf{f}}$ is the covariance matrix associated with the measurement noise given an object $\mathbf{f}$, $\langle \mathbf{K}_{\mathbf{n}|\mathbf{f}} \rangle_{\mathbf{f}|H_j}$ is the mean of $\mathbf{K}_{\mathbf{n}|\mathbf{f}}$ under the hypothesis H_j, and $\mathbf{K}_{\bar{\mathbf{g}}(\mathbf{f})|H_j} = \langle [\bar{\mathbf{g}}(\mathbf{f}) - \bar{\bar{\mathbf{g}}}_j][\bar{\mathbf{g}}(\mathbf{f}) - \bar{\bar{\mathbf{g}}}_j]^T \rangle_{\mathbf{f}|H_j}$ is the covariance matrix associated with the object $\mathbf{f}$ under the hypothesis H_j.

The signal-to-noise ratio associated with a test statistic, denoted as SNR_t, is a commonly used FOM to assess observer performance at signal detection tasks. The SNR_t is defined as [9]:

$$\mathrm{SNR}_t = \frac{\langle t\rangle_1 - \langle t\rangle_0}{\sqrt{\frac{1}{2}\sigma_0^2 + \frac{1}{2}\sigma_1^2}}, \tag{2.20}$$

where $\langle t\rangle_j$ is the mean of a test statistic t under the hypothesis H_j and $\sigma_j^2 = \left\langle (t - \langle t\rangle_j)^2 \right\rangle_j$ is the variance of t under the hypothesis H_j. The HO maximizes the value of SNR_t among all linear observers that can be computed as [9]:

$$\mathrm{SNR}_{HO}^2 = \Delta\bar{\bar{\mathbf{g}}}^T \mathbf{w}_{\mathrm{HO}}. \tag{2.21}$$

When the likelihood function $p(\mathbf{g}|H_0)$ and $p(\mathbf{g}|H_1)$ can be described by a Gaussian probability density function that have the same covariance matrix, i.e., $\mathbf{K}_0 = \mathbf{K}_1$, the HO is equivalent to the IO.

2.3 Joint signal detection and localization tasks

When a signal location is modeled as a discrete parameter having J possible values, a signal detection-localization task requires an observer to classify a measured image data $\mathbf{g}$ as satisfying either a signal-absent hypothesis H_0 or one of the signal-present hypotheses H_j that corresponds to the j^{th} signal location $(j = 1, 2, ..., J)$ [62]. The imaging processes under these hypotheses can be described as:

$$\begin{aligned} &H_0 : \mathbf{g} = \mathbf{b} + \mathbf{n}, \\ &H_j : \mathbf{g} = \mathbf{b} + \mathbf{s}_j + \mathbf{n}, \end{aligned} \tag{2.22}$$

where $\mathbf{b} \equiv \mathcal{H} f_b(\mathbf{r})$ is the image of a background $f_b(\mathbf{r})$ and $\mathbf{s}_j \equiv \mathcal{H} f_{s_j}(\mathbf{r})$ is the image of a signal $f_{s_j}(\mathbf{r})$ at the j^{th} location $(j = 1, 2, ..., J)$. Denote the m^{th} element of $\mathbf{b}$ and $\mathbf{s}_j$ as b_m

and s_{j_m}, respectively. These quantities can be computed as:

$$\begin{aligned} b_m &= \int_{\mathbb{R}^d} d\mathbf{r}\ h_m(\mathbf{r}) f_b(\mathbf{r}), \\ s_{j_m} &= \int_{\mathbb{R}^d} d\mathbf{r}\ h_m(\mathbf{r}) f_{s_j}(\mathbf{r}). \end{aligned} \tag{2.23}$$

Scanning observers are typically employed for performing signal detection-localization tasks [43]. A scanning observer computes a test statistic for each possible location and a decision can be subsequently made by employing a max-statistic rule. This decision strategy can be described as [43]:

$$\begin{gathered} t(\mathbf{g}) = \max_{j \in \{1,\ldots,J\}} \lambda_j(\mathbf{g}) \\ j^*(\mathbf{g}) = \arg\max_{j \in \{1,\ldots,J\}} \lambda_j(\mathbf{g}) \\ \text{Decide } H_{j^*(\mathbf{g})} \text{ if } t(\mathbf{g}) > \tau, \text{ else decide } H_0. \end{gathered} \tag{2.24}$$

Here, the function $\lambda_j(\mathbf{g})$ maps a measured image data $\mathbf{g}$ to a real-valued test statistic corresponding to the j^{th} signal location. The maximum test statistic among the J test statistics that correspond to the J possible signal locations is compared to a pre-determined threshold τ to make a decision. A localization receiver operating characteristic (LROC) curve that depicts the tradeoff between the probability of correct localization and the false-positive rate can be plotted by varying the threshold τ. The observer performance can be subsequently quantified by the area under the LROC curve (ALROC). A two-alternative forced-choice (2AFC) test can also be employed to compute the ALROC without plotting the LROC curve [24].

2.3.1 Scanning Ideal Observer

The scanning IO that employs a modified generalized likelihood ratio test (MGLRT) maximizes the observer performance at signal detection-localization tasks as measured by the ALROC [62]. The MGLRT can be represented as [62]:

$$\begin{aligned} t_{LR}(\mathbf{g}) &= \max_{j \in \{1,\ldots,J\}} \frac{\Pr(H_j)p(\mathbf{g}|H_j)}{p(\mathbf{g}|H_0)} \\ j^*_{LR}(\mathbf{g}) &= \arg\max_{j \in \{1,\ldots,J\}} \frac{\Pr(H_j)p(\mathbf{g}|H_j)}{p(\mathbf{g}|H_0)} \\ &\text{Decide } H_{j^*_{LR}(\mathbf{g})} \text{ if } t_{LR}(\mathbf{g}) > \tau_{LR}, \text{ else decide } H_0. \end{aligned} \tag{2.25}$$

It is useful to note that, according to Bayes rule, the MGLRT described in Eq. (2.25) is equivalent to a posterior ratio test [117]:

$$\begin{aligned} t_{PR}(\mathbf{g}) &= \max_{j \in \{1,\ldots,J\}} \frac{\Pr(H_j|\mathbf{g})}{\Pr(H_0|\mathbf{g})} \\ j^*_{PR}(\mathbf{g}) &= \arg\max_{j \in \{1,\ldots,J\}} \frac{\Pr(H_j|\mathbf{g})}{\Pr(H_0|\mathbf{g})} \\ &\text{Decide } H_{j^*_{PR}(\mathbf{g})} \text{ if } t_{PR}(\mathbf{g}) > \tau_{PR}, \text{ else decide } H_0. \end{aligned} \tag{2.26}$$

In the case where threshold τ_{PR} is set to one, the probability of error is minimized, and the corresponding decision rule is called minimum-error criterion or the maximum a posterior (MAP) criterion [9]. In addition, a possible IO test statistic for the simplified binary signal detection task can be computed as a posterior probability $\Pr(H_{present}|\mathbf{g})$ that describes the probability of the signal-present hypothesis $H_{present}$ given a measured image data $\mathbf{g}$:

$$\Pr(H_{present}|\mathbf{g}) = \sum_{j=1}^{J} \Pr(H_j|\mathbf{g}) \equiv 1 - \Pr(H_0|\mathbf{g}). \tag{2.27}$$

When the computation of the IO test statistic is analytically intractable, the MCMC method can be employed for some SOMs that describe the background as: $\mathbf{b} = \mathbf{b}(\boldsymbol{\theta})$. The likelihood ratio $\Lambda_j(\mathbf{g}) \equiv \frac{p(\mathbf{g}|H_j)}{p(\mathbf{g}|H_0)}$ can be approximated as [68]:

$$\hat{\Lambda}_j(\mathbf{g}) = \frac{1}{N_c} \sum_{i=1}^{N_c} \Lambda_{\mathrm{BKE}_j}(\mathbf{g}|\mathbf{b}(\boldsymbol{\theta}^i)), \tag{2.28}$$

where $\Lambda_{\mathrm{BKE}_j}(\mathbf{g}|\mathbf{b}(\boldsymbol{\theta}^i)) = \frac{p(\mathbf{g}|\mathbf{b}(\boldsymbol{\theta}^i),H_j)}{p(\mathbf{g}|\mathbf{b}(\boldsymbol{\theta}^i),H_0)}$ is the likelihood ratio corresponding to the j^{th} signal location conditioned on a background $\mathbf{b}(\boldsymbol{\theta}^i)$. The vector $\boldsymbol{\theta}^i$ can be sampled from a posterior distribution $p(\mathbf{b}|\mathbf{g}, H_0)$ by constructing a Markov chain using a Metropolis-Hastings algorithm as discussed in Section 2.2.1.

2.3.2 Scanning Hotelling Observer

When the likelihood $p(\mathbf{g}|H_j)$ can be described by a Gaussian probability densify function having the covariance matrix $\mathbf{K}$ under each hypothesis H_j $(j = 0, 1, 2, ..., J)$, the scanning IO is equivalent to the scanning HO [10, 41, 43]. Let $\bar{\mathbf{b}}$ denote the mean of background images: $\bar{\mathbf{b}} = \langle \mathbf{b} \rangle_{\mathbf{b}}$. When the prior probability $\Pr(H_j)$ is a constant, the scanning HO can be computed as:

$$\begin{aligned} t_{HO}(\mathbf{g}) &= \max_{j \in \{1,...,J\}} \mathbf{w}_{HO_j}^T \left(\mathbf{g} - \bar{\mathbf{b}} - \frac{\mathbf{s}_j}{2} \right) \\ j^*_{HO}(\mathbf{g}) &= \arg\max_{j \in \{1,...,J\}} \mathbf{w}_{HO_j}^T \left(\mathbf{g} - \bar{\mathbf{b}} - \frac{\mathbf{s}_j}{2} \right) \end{aligned} \tag{2.29}$$

$$\text{Decide } H_{j^*_{HO}(\mathbf{g})} \text{ if } t_{HO}(\mathbf{g}) > \tau_{HO}, \text{ else decide } H_0,$$

where $\mathbf{w}_{HO_j} = \mathbf{K}^{-1}\mathbf{s}_j$ is the Hotelling template corresponding to the j^{th} signal location. Because the computation of the scanning HO is relatively easy, it can be employed to assess and optimize imaging systems and data-acquisition designs when the scanning IO is intractable or difficult to compute.

It should be noted that the scanning observers described above correspond to a discrete signal location model in which the search tolerance is not involved. Detailed discussions on the search tolerance can be found in [62]. The considered scanning observers can be generalized to the signal location models that involve search tolerances according to [62]. In this dissertation, we will focus on the discrete signal location model without search tolerances.

Chapter 3

Approximating the Ideal Observer and the Hotelling Observer for binary signal detection tasks by use of supervised learning methods

3.1 Overview

When binary signal detection tasks are considered for assessing and optimizing imaging systems and data-acquisition designs, the performance of the Ideal Observer (IO) has been advocated for use as a figure-of-merit (FOM). As introduced in Chapter 2, the IO test statistic involves the likelihood ratio that is intractable to compute in the majority cases. Sampling-based methods that employ Markov-Chain Monte Carlo (MCMC) techniques have been proposed to address this difficulty [2, 68]. However, to properly implement MCMC methods, many practical issues such as the design of proposal density function from which Markov chains can be efficiently constructed need to be addressed. Current applications of MCMC methods have been limited to some relatively simple object models such lumpy object models [68], binary texture models [2] and parameterized torso phantoms [50]. The Hotelling observer (HO), which the optimal linear observer, can be employed to assess objective measures of IQ when the IO is difficult or intractable to compute. However, as discussed in Chapter 2, computation of the HO can be challenging because a potentially large covariance matrix needs to be estimated and subsequently inverted. These computational challenges limit the utilization of objective measures of IQ for assessing and optimizing imaging systems and data-acquisition designs.

Supervised learning methods that employ artificial neural networks (ANNs) hold great promise for the design and implementation of model observers [3, 16, 75, 107]. When optimizing imaging systems and data-acquisition designs, computer-simulation data are commonly employed. In such applications, large amounts of data can be generated for training complicated inference models that are represented by ANNs. Kupinski *et al.* have demonstrated the ability of fully-connected neural networks (FCNNs) to approximate the IO that acts on low-dimensional image feature vectors [67]. However, because each neuron in FCNNs is connected to all neurons in the previous layer, the dimensionality of the input layer and the depth of the FCNN that can be effectively trained are limited. Therefore, FCNNs do not scale well to high dimensional image data.

Modern deep learning methods that employ convolutional neural networks (CNNs) have been developed to address this limitation [22, 40, 70, 71]. In this chapter, inspired by the success of CNNs in image classification tasks, we propose and investigate a supervised learning method that employs CNNs to approximate the IO that directly acts on image data for binary signal detection tasks. Novel supervised learning methods that employ single layer neural networks (SLNNs) to approximate the HO are also provided. The proposed methods directly estimate the Hotelling template without explicitly estimating and inverting covariance matrices. Therefore, they can be used for computing the HO for large images.

3.2 Approximating the IO for signal detection tasks by use of convolutional neural networks (CNNs)

According to Eq. (2.6), the IO test statistic can be computed as a posterior probability $\Pr(H_1|\mathbf{g})$, which is a monotonic transformation of the likelihood ratio. To train a CNN for approximating $\Pr(H_1|\mathbf{g})$, a sigmoid function is employed in the last layer of the CNN. In this way, the output of the CNN can be interpreted as a probability. Denote a set of all weight parameters of a CNN as $\boldsymbol{\Theta}$. Let the probability represented by the CNN be denoted as $\Pr(H_1|\mathbf{g}, \boldsymbol{\Theta})$. The vector $\boldsymbol{\Theta}$ is determined in the training of the CNN such that the difference between the actual probability $\Pr(H_1|\mathbf{g})$ and the CNN-represented probability $\Pr(H_1|\mathbf{g}, \boldsymbol{\Theta})$ is small. The probability $\Pr(H_0|\mathbf{g})$ can be subsequently approximated by $\Pr(H_0|\mathbf{g}, \boldsymbol{\Theta}) \equiv 1 - \Pr(H_1|\mathbf{g}, \boldsymbol{\Theta})$.

The maximum likelihood (ML) estimate of the CNN parameters $\boldsymbol{\Theta}$ can be obtained by use of a supervised learning method. Denote the label of the measured image data $\mathbf{g}$ as $y \in \{0, 1\}$, where $y = 0$ corresponds to the signal-absent hypothesis H_0 and $y = 1$ corresponds to the signal-present hypothesis H_1. Given the joint probability distribution $p(\mathbf{g}, H_y)$, the ML estimate of the CNN parameters $\boldsymbol{\Theta}$ can be obtained by minimizing the generalization error, which is defined as the ensemble average of the cross-entropy over $p(\mathbf{g}, H_y)$:

$$\boldsymbol{\Theta}_{\mathrm{ML}} = \arg\min_{\boldsymbol{\Theta}} \left\langle - \log \left[\Pr(H_y|\mathbf{g}, \boldsymbol{\Theta}) \right] \right\rangle_{(\mathbf{g}, H_y)}, \tag{3.1}$$

where $\langle . \rangle_{(\mathbf{g},H_y)}$ represents the average over the joint probability distribution $p(\mathbf{g}, H_y)$. If the CNN is sufficiently complex to represent any functional form, $\Pr(H_1|\mathbf{g}, \mathbf{\Theta}_{\mathrm{ML}}) = \Pr(H_1|\mathbf{g})$ when the global minimum of Eq. (3.1) is achieved. To see this, one can compute the generalization error as:

$$\begin{aligned}&\left\langle -\log\left[\Pr(H_y|\mathbf{g},\mathbf{\Theta})\right]\right\rangle_{(\mathbf{g},H_y)}\\&= -\int_{\mathbb{R}^M}\Big[\log\big(\Pr(H_1|\mathbf{g},\mathbf{\Theta})\big)p(\mathbf{g},H_1) + \log\big(1-\Pr(H_1|\mathbf{g},\mathbf{\Theta})\big)p(\mathbf{g},H_0)\Big]d^M\mathbf{g}.\end{aligned} \tag{3.2}$$

The cross-entropy $\left\langle -\log\left[\Pr(H_y|\mathbf{g},\mathbf{\Theta})\right]\right\rangle_{(\mathbf{g},H_y)}$ can be considered as a functional of $\Pr(H_1|\mathbf{g}, \mathbf{\Theta})$. Here, $\Pr(H_1|\mathbf{g}, \mathbf{\Theta})$ is a function of $\mathbf{g}$. The functional derivative of $\left\langle -\log\left[\Pr(H_y|\mathbf{g},\mathbf{\Theta})\right]\right\rangle_{(\mathbf{g},H_y)}$ with respect to $\Pr(H_1|\mathbf{g}, \mathbf{\Theta})$, which is known as a Fréchet derivative, can be computed as:

$$\frac{\partial\left\langle -\log\left[\Pr(H_y|\mathbf{g},\mathbf{\Theta})\right]\right\rangle_{(\mathbf{g},H_y)}}{\partial \Pr(H_1|\mathbf{g},\mathbf{\Theta})} = -\left[\frac{\Pr(H_1|\mathbf{g})}{\Pr(H_1|\mathbf{g},\mathbf{\Theta})} - \frac{1-\Pr(H_1|\mathbf{g})}{1-\Pr(H_1|\mathbf{g},\mathbf{\Theta})}\right]p(\mathbf{g}). \tag{3.3}$$

For any $\mathbf{g} \in \{\mathbf{g}|p(\mathbf{g}) \neq 0\}$, the derivative in Eq. (3.3) equals zero when $\frac{\Pr(H_1|\mathbf{g})}{\Pr(H_1|\mathbf{g},\mathbf{\Theta})} = \frac{1-\Pr(H_1|\mathbf{g})}{1-\Pr(H_1|\mathbf{g},\mathbf{\Theta})}$, from which $\Pr(H_1|\mathbf{g}, \mathbf{\Theta}_{\mathrm{ML}}) = \Pr(H_1|\mathbf{g})$.

Given a set of training data that comprises N labeled images $\{(\mathbf{g}_i, y_i)\}_{i=1}^N$, $\mathbf{\Theta}_{\mathrm{ML}}$ can be estimated by minimizing the empirical error function, which is defined as the average of the cross-entropy over the training dataset:

$$\hat{\mathbf{\Theta}}_{\mathrm{ML}} = \arg\min_{\mathbf{\Theta}}\left[-\frac{1}{N}\sum_{i=1}^{N}\log\big(\Pr(H_{y_i}|\mathbf{g}_i,\mathbf{\Theta})\big)\right], \tag{3.4}$$

where $\hat{\mathbf{\Theta}}_{\mathrm{ML}}$ is an empirical estimate of $\mathbf{\Theta}_{\mathrm{ML}}$. It should be noted that minimizing empirical errors on small dataset can cause overfitting and large generalization errors [45]. To reduce the rate at which overfitting happens, mini-batch stochastic gradient descent algorithms can be employed. When online learning is implemented, mini-batches are drawn on-the-fly from the joint distribution $p(\mathbf{g}, H_y)$ [45].

3.3 Approximating the HO by use of single layer neural networks

The use of SLNNs to approximate the HO is presented in this section.

3.3.1 Training the HO by use of supervised learning method

As discussed in Chapter 2, the HO implements a test statistic that is a linear function of the measured image data $\mathbf{g}$. To represent a linear function, a single-layer neural network (SLNN) that has a linear fully connected layer can be employed. Denote the vector of SLNN weight parameters as $\mathbf{w} \in \mathbb{R}^{M\times 1}$. The test statistic computed by a SLNN can be represented as:

$$t_{\text{SLNN}}(\mathbf{g}) = \mathbf{w}^T\mathbf{g}. \tag{3.5}$$

To approximate the HO, the SLNN is trained to maximize SNR_t by solving the following optimization problem:

$$\begin{aligned} \underset{\mathbf{w}}{\text{minimize}} \quad & \frac{1}{2}\left\langle[\mathbf{w}^T\mathbf{g} - \mathbf{w}^T\bar{\bar{\mathbf{g}}}_0]^2\right\rangle_0 + \frac{1}{2}\left\langle[\mathbf{w}^T\mathbf{g} - \mathbf{w}^T\bar{\bar{\mathbf{g}}}_1]^2\right\rangle_1 \\ \text{subject to} \quad & \mathbf{w}^T\bar{\bar{\mathbf{g}}}_1 - \mathbf{w}^T\bar{\bar{\mathbf{g}}}_0 = C, \end{aligned} \tag{3.6}$$

where C is a real-valued positive scalar. The Lagrangian function associated with this constrained optimization problem can be computed as:

$$L(\mathbf{w}, \lambda) = \frac{1}{2}\left\langle[\mathbf{w}^T\mathbf{g} - \mathbf{w}^T\bar{\bar{\mathbf{g}}}_0]^2\right\rangle_0 + \frac{1}{2}\left\langle[\mathbf{w}^T\mathbf{g} - \mathbf{w}^T\bar{\bar{\mathbf{g}}}_1]^2\right\rangle_1 - \lambda(\mathbf{w}^T\bar{\bar{\mathbf{g}}}_1 - \mathbf{w}^T\bar{\bar{\mathbf{g}}}_0 - C). \tag{3.7}$$

The optimal solution $\mathbf{w}^*$ satisfies the Lagrange multiplier conditions that can be represented as:

$$\nabla_{\mathbf{w}} L(\mathbf{w}^*, \lambda^*) = [\mathbf{K}_0 + \mathbf{K}_1]\, \mathbf{w}^* - \lambda^* \Delta\bar{\bar{\mathbf{g}}} = 0, \tag{3.8a}$$

$$\nabla_{\lambda} L(\mathbf{w}^*, \lambda^*) = -\left[\mathbf{w}^{*T} \Delta\bar{\bar{\mathbf{g}}} - C\right] = 0, \tag{3.8b}$$

where λ^* denotes the Lagrange multiplier. According to Eq. (3.8), $\mathbf{w}^*$ and λ^* can be computed as:

$$\mathbf{w}^* = \left[\frac{1}{\lambda^*}(\mathbf{K}_0 + \mathbf{K}_1)\right]^{-1} \Delta\bar{\bar{\mathbf{g}}}, \tag{3.9a}$$

$$\lambda^* = \frac{C}{\Delta\bar{\bar{\mathbf{g}}}^T (\mathbf{K}_0 + \mathbf{K}_1)^{-1} \Delta\bar{\bar{\mathbf{g}}}}. \tag{3.9b}$$

Because the Lagrangian function in Eq. (3.7) is convex, $\mathbf{w}^*$ is the global minimum of $L(\mathbf{w}, \lambda^*)$ and the constrained optimization problem defined in Eq. (3.6) can be solved by minimizing $L(\mathbf{w}, \lambda^*)$ with respect to $\mathbf{w}$. In addition, because minimizing $L(\mathbf{w}, \lambda^*)$ with respect to $\mathbf{w}$ is equivalent to minimizing $L(\mathbf{w}, \lambda^*) - \lambda^* C$ with respect to $\mathbf{w}$, the generalization error to be minimized is defined as:

$$\begin{aligned} l(\mathbf{w}) &= L(\mathbf{w}, \lambda^*) - \lambda^* C \\ &= \frac{1}{2}\left\langle [\mathbf{w}^T(\mathbf{g} - \bar{\bar{\mathbf{g}}}_0)]^2 \right\rangle_0 + \frac{1}{2}\left\langle [\mathbf{w}^T(\mathbf{g} - \bar{\bar{\mathbf{g}}}_1)]^2 \right\rangle_1 - \lambda^* \mathbf{w}^T \Delta\bar{\bar{\mathbf{g}}}. \end{aligned} \tag{3.10}$$

In order to have $\mathbf{w}^* = \mathbf{w}_{\mathrm{HO}}$, λ^* is set to 2.

Given a set of training image data $\{\mathbf{g}_i, y_i\}_{i=1}^N$ in which half of them are signal-absent and half of them are signal-present, $\mathbf{w}_{\mathrm{HO}}$ can be estimated by minimizing the empirical error:

$$\hat{l}(\mathbf{w}) = \frac{1}{N}\sum_{i=1}^{N} \left\{ (1 - y_i)\left[\mathbf{w}^T(\mathbf{g}_i - \hat{\mathbf{g}}_0)\right]^2 + y_i \left[\mathbf{w}^T(\mathbf{g}_i - \hat{\mathbf{g}}_1)\right]^2 \right\} - 2\mathbf{w}^T \Delta\hat{\mathbf{g}}, \tag{3.11}$$

where $\hat{\mathbf{g}}_0 = \frac{2}{N}\sum_{i=1}^{N}(1-y_i)\mathbf{g}_i$, $\hat{\mathbf{g}}_1 = \frac{2}{N}\sum_{i=1}^{N} y_i\mathbf{g}_i$, and $\Delta\hat{\mathbf{g}} = \hat{\mathbf{g}}_1 - \hat{\mathbf{g}}_0$. Any gradient-based algorithm can be employed to minimize the empirical error $\hat{l}(\mathbf{w})$ to estimate the Hotelling template. Because this method does not need to explicitly estimate and invert the covariance matrix, it can scale well to large images.

3.3.2 Learning the HO by use of a Covariance-Matrix Decomposition

Methods that employ a covariance-matrix decomposition have been developed to estimate and invert covariance matrices for computing the Hotelling template [9, 65]. As discussed in Eq. (2.19), the covariance matrix $\mathbf{K}_j$ associated with the hypothesis H_j $(j = 0, 1)$ can be decomposed as: $\mathbf{K}_j = \langle \mathbf{K}_{\mathbf{n}|\mathbf{f}} \rangle_{\mathbf{f}|H_j} + \mathbf{K}_{\bar{\mathbf{g}}(\mathbf{f})|H_j}$. When computer-simulation is conducted for assessing and optimizing imaging systems, the covariance matrix associated with the noise $\mathbf{K}_{\mathbf{n}|\mathbf{f}}$ is known. When uncorrelated noise is considered, $\langle \mathbf{K}_{\mathbf{n}|\mathbf{f}} \rangle_{\mathbf{f}|H_j}$ is a diagonal matrix. In cases where detectors introduce correlations in the measurement data, $\langle \mathbf{K}_{\mathbf{n}|\mathbf{f}} \rangle_{\mathbf{f}|H_j}$ may be a banded and nearly diagonal matrix. In this section, a novel supervised learning method is proposed to approximate the HO by use of a covariance-matrix decomposition.

According to the covariance-matrix decomposition described in Eq. (2.19), the variance of the test statistic under the hypothesis H_j can be computed as:

$$\left\langle (\mathbf{w}^T\mathbf{g} - \langle \mathbf{w}^T\mathbf{g} \rangle_j)^2 \right\rangle_j = \mathbf{w}^T \mathbf{K}_{\bar{\mathbf{g}}(\mathbf{f})|H_j} \mathbf{w} + \mathbf{w}^T \langle \mathbf{K}_{\mathbf{n}|\mathbf{f}} \rangle_{\mathbf{f}|H_j} \mathbf{w}. \tag{3.12}$$

Let $\overline{\mathbf{K}}_{\mathbf{n}}$ denote the averaged covariance matrix associated with the noise, which is assumed known:

$$\overline{\mathbf{K}}_{\mathbf{n}} \equiv \frac{1}{2}(\langle \mathbf{K}_{\mathbf{n}|\mathbf{f}} \rangle_{\mathbf{f}|H_0} + \langle \mathbf{K}_{\mathbf{n}|\mathbf{f}} \rangle_{\mathbf{f}|H_1}). \tag{3.13}$$

The generalization error defined in Eq. (3.10) can be written as:

$$l(\mathbf{w}) = \left\langle (\mathbf{w}^T\mathbf{b} - \mathbf{w}^T\bar{\mathbf{b}})^2 \right\rangle_{\mathbf{f}_b} + \frac{1}{2}\left\langle (\mathbf{w}^T\mathbf{s} - \mathbf{w}^T\bar{\mathbf{s}})^2 \right\rangle_{\mathbf{f}_s} + \mathbf{w}^T\overline{\mathbf{K}}_\mathbf{n}\mathbf{w} - 2\mathbf{w}^T\bar{\mathbf{s}}, \tag{3.14}$$

where $\bar{\mathbf{b}} = \langle \mathbf{b} \rangle_{\mathbf{f}_b}$, and $\bar{\mathbf{s}} = \langle \mathbf{s} \rangle_{\mathbf{f}_s}$.

Given a set of background images $\{\mathbf{b}_i\}_{i=1}^N$ and a set of signal images $\{\mathbf{s}_i\}_{i=1}^N$, the empirical error to be minimized is:

$$\hat{l}(\mathbf{w}) = \frac{1}{N}\sum_{i=1}^{N}\left\{[\mathbf{w}^T\mathbf{b}_i - \mathbf{w}^T\hat{\mathbf{b}}]^2 + \frac{1}{2}[\mathbf{w}^T\mathbf{s}_i - \mathbf{w}^T\hat{\mathbf{s}}]^2\right\} + \mathbf{w}^T\overline{\mathbf{K}}_\mathbf{n}\mathbf{w} - 2\mathbf{w}^T\hat{\mathbf{s}}, \tag{3.15}$$

where $\hat{\mathbf{b}} = \frac{1}{N}\sum_{i=1}^N \mathbf{b}_i$, and $\hat{\mathbf{s}} = \frac{1}{N}\sum_{i=1}^N \mathbf{s}_i$. Any gradient-based algorithms can be employed to minimize the empirical error defined in Eq. (3.15) to estimate the Hotelling template. This method also does not need to explicitly invert covariance matrix and therefore, it can scale well to large images.

3.4 Numerical studies

To investigate and validate the proposed supervised learning methods for approximating the IO and HO test statistics, computer-simulation studies that involve four different binary signal detection tasks were conducted. A *signal-known-exactly and background-known-exactly* (SKE/BKE) signal detection task was considered in which the IO and HO can be analytically computed. A *signal-known-exactly and background-known-statistically* (SKE/BKS) detection task and a *signal-known-statistically and background-known-statistically* (SKS/BKS) detection task were considered in which the background is modeled by a lumpy object model [91]. For these two BKS signal detection tasks that employ a lumpy object model, the IO test statistic can be computed by use of MCMC methods [68, 85]. Finally, a SKE/BKS signal detection task that employs a clustered lumpy background (CLB) object model [13] was considered. To the best of our knowledge, current applications of the MCMC methods to CLB object model have not been reported [2]. For all considered binary signal detection tasks, the observer performance was assessed by use of the ROC curve that was fit by use of the Metz-ROC software [76] that utilized the "proper" binormal model [77, 87].

The imaging system in this study was simulated by a linear C-D mapping with a Gaussian kernel that was motivated by an idealized parallel-hole collimator system [66, 68]:

$$h_m(\mathbf{r}) = \frac{h}{2\pi w^2} \exp\left(\frac{-\left(\mathbf{r} - \mathbf{r}_m\right)^T \left(\mathbf{r} - \mathbf{r}_m\right)}{2w^2} \right), \tag{3.16}$$

where the height $h = 40$ and the width $w = 0.5$. The details for each binary signal detection task and the training of neural networks are given in the following subsections.

3.4.1 Signal-known-exactly (SKE) and background-known-exactly (BKE) signal detection task

In this case, both the signal and background were non-random. The image size was 64×64 and the background image was specified as $\mathbf{b} = \mathbf{0}$. The signal function $f_s(\mathbf{r})$ was a 2D symmetric Gaussian function:

$$f_s(\mathbf{r}) = A \exp\left(\frac{-\left(\mathbf{r} - \mathbf{r}_c\right)^T \left(\mathbf{r} - \mathbf{r}_c\right)}{2w_s^2} \right), \tag{3.17}$$

where $A = 0.2$ is the amplitude, $\mathbf{r}_c = [32, 32]^T$ is the coordinate of the signal location, and $w_s = 3$ is the width of the signal. The signal image $\mathbf{s}$ can be computed as:

$$s_m = \frac{A h w_s^2}{(w^2 + w_s^2)} \exp\left(-\frac{(\mathbf{r}_m - \mathbf{r}_c)^T (\mathbf{r}_m - \mathbf{r}_c)}{2(w^2 + w_s^2)} \right). \tag{3.18}$$

To demonstrate the ability of CNNs to approximate a non-linear IO test statistic, a Laplacian distribution that has been employed to describe fine details of mammographic images was considered [25]. Specifically, the measurement data $\mathbf{g}$ were simulated by adding independent and identically distributed (i.i.d.) Laplacian noise: $n_m \sim \mathcal{L}(0, c)$, where $\mathcal{L}(0, c)$ denotes a Laplacian distribution with the mean of 0 and the exponential decay of c. The value of c was set to $30/\sqrt{2}$ that corresponds to standard deviation 30.

The IO test statistic for this case can be computed as [25]:

$$\Lambda(\mathbf{g}) = \exp\left[\frac{1}{c}\sum_{m=1}^{M}(|g_m - b_m| - |g_m - b_m - s_m|)\right]. \tag{3.19}$$

The Hotelling template can be computed by analytically inverting the covariance matrix $\mathbf{K}_j \in \mathbb{R}^{M \times M}$ $(j = 0, 1)$:

$$\mathbf{K}_j^{-1}(m, n) = \begin{cases} \frac{1}{2c^2}, & \text{if } m = n \\ 0, & \text{if } m \neq n, \end{cases} \tag{3.20}$$

where $\mathbf{K}_j^{-1}(m, n)$ denotes the component at the m^{th} row and the n^{th} column $(1 \leq m, n \leq M)$ of $\mathbf{K}_j^{-1}$. The observer performances produced by the proposed supervised learning methods were compared to those produced by the analytical computations as described above.

3.4.2 SKE and background-known-statistically (BKS) signal detection task with a lumpy background model

The image size was 64 × 64 for this case. A non-random signal described by Eq. (3.17) was employed and the background was modeled by a stochastic lumpy object model [91]:

$$f_b(\mathbf{r}) = \sum_{n=1}^{N_b} l(\mathbf{r} - \mathbf{r}_n | a, s), \tag{3.21}$$

where N_b is the number of lumps that is drawn from Poisson distribution with the mean $\overline{N}$: $N_b \sim \mathcal{P}(\overline{N})$. Here, $\mathcal{P}(\overline{N})$ denotes a Poisson distribution with the mean $\overline{N}$, which was set to 5, and $l(\mathbf{r} - \mathbf{r}_n | a, s)$ is a lumpy function that was modeled by a 2D Gaussian function with the amplitude a and width s:

$$l(\mathbf{r} - \mathbf{r}_n | a, s) = a \exp\left(- \frac{(\mathbf{r} - \mathbf{r}_n)^T (\mathbf{r} - \mathbf{r}_n)}{2s^2} \right). \tag{3.22}$$

Here, a was set to 1, s was set to 7, and $\mathbf{r}_n$ is the location of the n^{th} lump that was drawn from uniform distribution over the field of view. The background image $\mathbf{b}$ was analytically computed as:

$$b_m = \frac{ahs^2}{w^2 + s^2} \sum_{n=1}^{N_b} \exp\left(- \frac{(\mathbf{r}_n - \mathbf{r}_m)^T (\mathbf{r}_n - \mathbf{r}_m)}{2(w^2 + s^2)} \right). \tag{3.23}$$

The measurement noise was an i.i.d. Gaussian noise that can model electronic noise: $n_m \sim \mathcal{N}(0, \delta^2)$, where $\mathcal{N}(0, \delta^2)$ denotes a Gaussian distribution with the mean 0 and the standard deviation δ, which was set to 20. Examples of signal-present images are shown in the top row of Fig. 3.1.

The IO and HO test statistics cannot be analytically determined for this case. The MCMC method was employed to serve as a surrogate for ground truth of the IO. In each Markov Chain, 200,000 background images were simulated according to the proposal density and the acceptance probability defined in [68]. The traditional HO test statistic was computed by use of the covariance-matrix decomposition [9] in which the background covariance matrix was empirically calculated by use of 100,000 background images.

3.4.3 Signal-known-statistically (SKS) and BKS signal detection task with a lumpy background model

This case employed the same stochastic lumpy background model that was specified in the SKE/BKS case described above. The signal was random and modeled by a 2D Gaussian function with a random location and a random shape, which can be mathematically represented as:

$$f_s(\mathbf{r}) = A \exp\Big(-\left[\mathbf{R}_\theta \left(\mathbf{r}-\mathbf{r}_c\right)\right]^T \mathbf{D}^{-1} \left[\mathbf{R}_\theta(\mathbf{r}-\mathbf{r}_c)\right]\Big). \tag{3.24}$$

Here, $\mathbf{R}_\theta = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}$ is a rotation matrix that rotates a vector through an angle θ in Euclidean space, and $\mathbf{D} = \begin{bmatrix} 2w_1^2 & 0 \\ 0 & 2w_2^2 \end{bmatrix}$ determines the width of the Gaussian function along each coordinate axis. The signal image $\mathbf{s}$ was analytically computed as:

$$s_m = A' \exp\Big(-\left[\mathbf{R}_\theta \left(\mathbf{r}_m-\mathbf{r}_c\right)\right]^T \mathbf{D}'^{-1} \left[\mathbf{R}_\theta(\mathbf{r}_m-\mathbf{r}_c)\right]\Big), \tag{3.25}$$

where $A' = Ahw_1w_2\sqrt{\frac{1}{(w^2+w_1^2)(w^2+w_2^2)}}$ and $\mathbf{D}' = \begin{bmatrix} 2(w^2+w_1^2) & 0 \\ 0 & 2(w^2+w_2^2) \end{bmatrix}$. The value of A was set to 0.2, θ was drawn from a uniform distribution: $\theta \sim \mathcal{U}(0, 2\pi)$, w_1 and w_2 were sampled from a uniform distribution: $w_1, w_2 \sim \mathcal{U}(2, 4)$, and $\mathbf{r}_c$ was uniformly distributed over the image field of view. The measurement noise was Gaussian having zero mean and a standard deviation of 10.

The MCMC method was employed to provide a surrogate for ground truth for the IO. In each Markov Chain, 400,000 background images were sampled according to the proposal density and the acceptance probability described in [85]. The traditional HO test statistic was calculated by use of the covariance-matrix decomposition [9] with an empirical object covariance matrix that was estimated by use of 100,000 background images and 100,000 signal images.

Because linear observers typically are unable to detect signals with random locations, the HO was expected to perform poorly. Multi-template model observers [20, 34, 115] and the scanning HO [10, 42] can be employed to detect variable signals. Sub-ensemble-based approaches have also been developed that can assess the performance on variable signal detection tasks [36, 74]. In this paper, we do not provide a method for training these observers. The approximation of these observers by use of a supervised learning method represents a topic for future investigation.

3.4.4 SKE and BKS signal detection task with a clustered lumpy background model (CLB)

Another SKE/BKS detection task associated with a more sophisticated stochastic background model, the clustered lumpy background (CLB), was considered also. The CLB model can be employed to synthesize images that resemble mammographic images [13]. In this study, the image size was set to 128 × 128 and a CLB realization was simulated as:

$$b_m = \sum_{k=1}^{K} \sum_{n=1}^{N_k} l\left(\mathbf{r}_m - \mathbf{r}_k - \mathbf{r}_{kn} | \mathrm{R}_{\theta_{kn}}\right), \tag{3.26}$$

where $K \sim \mathcal{P}(\overline{K})$ is the number of clusters, $N_k \sim \mathcal{P}(\overline{N})$ is the number of blobs in the k^{th} cluster, $\mathbf{r}_k$ is the location of the k^{th} cluster, and $\mathbf{r}_{kn}$ is the location of the n^{th} blob in the k^{th} cluster. Here, $\mathbf{r}_k$ was sampled from a uniform distribution over the image field of view, $\mathbf{r}_{kn}$ was sampled from a Gaussian distribution with standard deviation σ and center $\mathbf{r}_k$, and $l(\mathbf{r}|\mathrm{R}_{\theta_{kn}})$ is the blob function:

$$l(\mathbf{r}|\mathbf{R}_{\theta_{kn}}) = a \exp\left(-\alpha \frac{\|\mathbf{R}_{\theta_{kn}}\mathbf{r}\|^{\beta}}{\mathrm{L}(\mathbf{R}_{\theta_{kn}}\mathbf{r})}\right), \tag{3.27}$$

where a, α and β are adjustable parameters. The rotation matrix $\mathbf{R}_{\theta_{kn}}$ is associated with the angle $\theta_{kn} \sim \mathcal{U}(0, 2\pi)$, and $\mathrm{L}(\mathbf{r})$ is the "radius" of the ellipse with half-axes L_x and L_y:

$$\mathrm{L}(\mathbf{r}) = \frac{L_x L_y}{\sqrt{L_x^2 \sin^2(\theta_{\mathbf{r}}) + L_y^2 \cos^2(\theta_{\mathbf{r}})}}, \tag{3.28}$$

where $\theta_{\mathbf{r}} = \arctan(\frac{r_y}{r_x})$. Here, r_x and r_y denote the components of $\mathbf{r}$. The parameters employed for generating the CLB images are summarized in Table 3.1.

Table 3.1: Parameters for generating CLB images.

$\overline{\mathrm{K}}$	$\overline{\mathrm{N}}$	Lx	Ly	α	β	σ	a
150	20	5	2	2.1	0.5	12	100

The signal image was generated as a 2D symmetric Gaussian function centered in the image with an amplitude of 500 and a width of 12. Mixed Poisson-Gaussian noise that models both photon noise and electronic noise was employed. The standard deviation of Gaussian noise was set to 10. Examples of signal-present images are shown in the bottom row of Fig. 3.1.

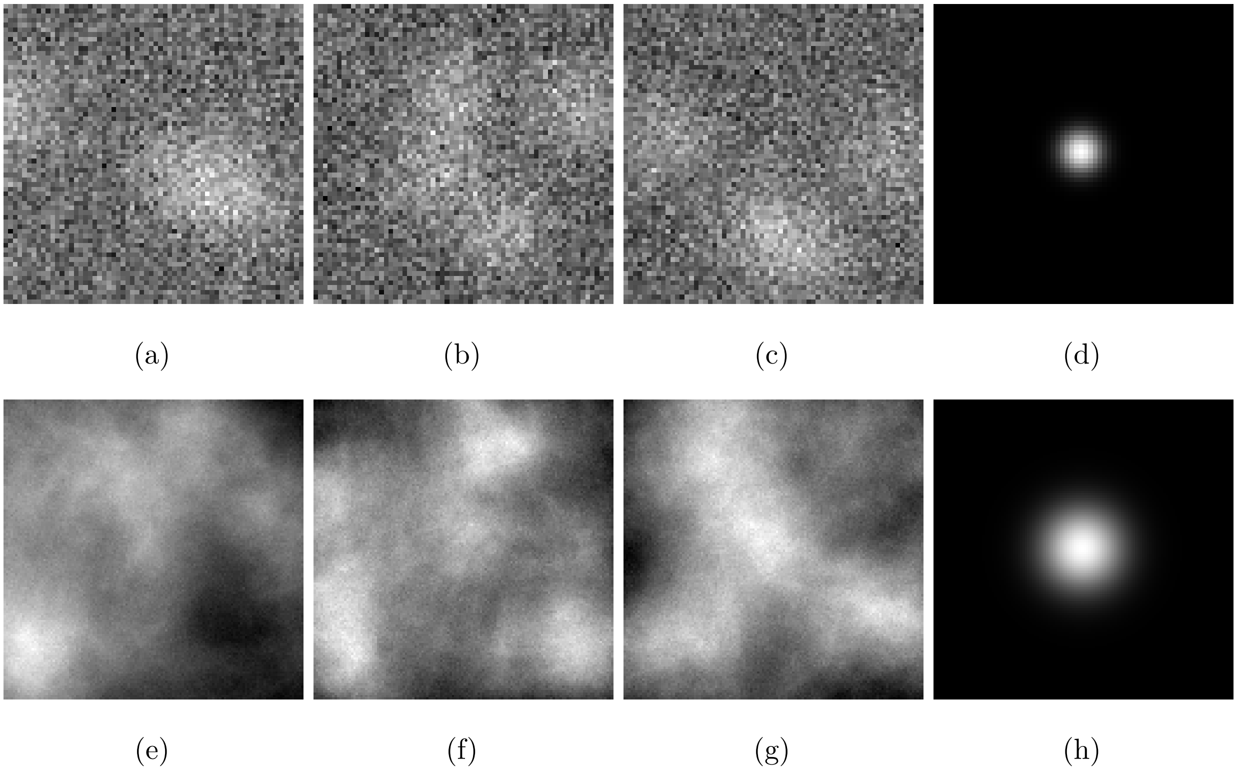


Figure 3.1: (a)-(c) Samples of the signal-present measurements for the SKE/BKS detection task with the lumpy background model. (d) An image showing the signal contained in (a)-(c). (e)-(g) Samples of the signal-present measurements for the SKE/BKS detection task with the CLB model. (h) An image showing the signal contained in (e)-(g). (© IEEE 2019)

To the best of our knowledge, current MCMC methods have not been applied to the CLB object model and the mixed Poisson-Gaussian noise model. To provide a surrogate for ground truth for the HO, the traditional HO was computed by use of covariance-matrix decomposition with the empirical background covariance matrix estimated using 400,000 background images.

3.4.5 Details of training neural networks

Here, details regarding the implementation of the supervised learning-based methods for approximating the IO and HO for the tasks above are described.

The train-validation-test scheme [45] was employed to evaluate the proposed supervised learning approaches. Specifically, the CNNs and SLNNs were trained on a training dataset. Subsequently, these neural networks were specified based upon a validation dataset and the detection performances of these networks were finally assessed on a testing dataset. To prepare training datasets for the BKS detection tasks, 100,000 lumpy background [91] images and 400,000 CLB images [13] were generated. When training the CNNs for approximating IOs, to mitigate the overfitting that can be caused by insufficient training data, a "semi-online learning" method was proposed and employed. In this approach, the measurement noise was generated on-the-fly and added to noiseless images drawn from the finite datasets. The validation dataset and testing dataset both comprised 200 images for each class.

To approximate the HO test statistic, SLNNs that represent linear functions were trained by use of the proposed method employing the covariance-matrix decomposition described in Eq. (3.15). This was possible because the noise models for the considered detection tasks were known. At each iteration in training processes, the parameters of SLNNs were updated by minimizing error function Eq. (3.15) on mini-batches drawn from the training dataset. Specifically, when training the SLNN for the SKE/BKE detection task, the signal and background that were known exactly were employed and each mini-batch contained the fixed signal image and background image. When training the SLNNs for the SKE/BKS detection tasks, the known signals were employed and each mini-batch contained 200 background images and the fixed signal image. For training the SLNN for the SKS/BKS detection task, each mini-batch contained 200 background images and 200 signal images. The weight vector $\mathbf{w}$ that produced the maximum SNR_t value evaluated on the validation dataset was specified to approximate the Hotelling template. The feasibility of the proposed methods for approximating the HO from a reduced number of images was also investigated. Specifically, the SLNNs were trained for the SKE/BKS detection task with the CLB model by minimizing Eq. (3.11) and Eq. (3.15) on datasets comprising 2000 labeled measurements (contained 1000 signal-present images and 1000 signal-absent images) and 2000 background images, respectively.

As opposed to the case of the HO approximation where the network architecture is known linear, to specify the CNN architecture for approximating the IO, a family of CNNs that possess different numbers of convolutional (CONV) layers was explored. Specifically, an initial CNN having one CONV layer was firstly trained by minimizing the cross-entropy described in Eq. (3.4). Subsequently, CNNs having additional CONV layers were trained according to Eq. (3.4) until the network did not significantly decrease the cross-entropy on a validation dataset. The cross-entropy was considered as significantly decreased if its decrement is at least 1.0% of that produced by the previous CNN. Finally the CNN having the minimum validation cross-entropy was selected as the optimal CNN in the explored architecture family. For all the considered CNN architectures in this architecture family, each CONV layer comprised 32 filters with 5×5 spatial support and was followed by a LeakyReLU activation function [99], a max-pooling layer [93] following the last CONV layer was employed to subsample the feature maps, and finally a fully connected (FC) layer using a sigmoid activation function computed the posterior probability $\Pr(H_1|\mathbf{g}, \boldsymbol{\Theta})$. It should be noted that these architecture parameters were determined heuristically and may not be optimal for many signal detection tasks. One

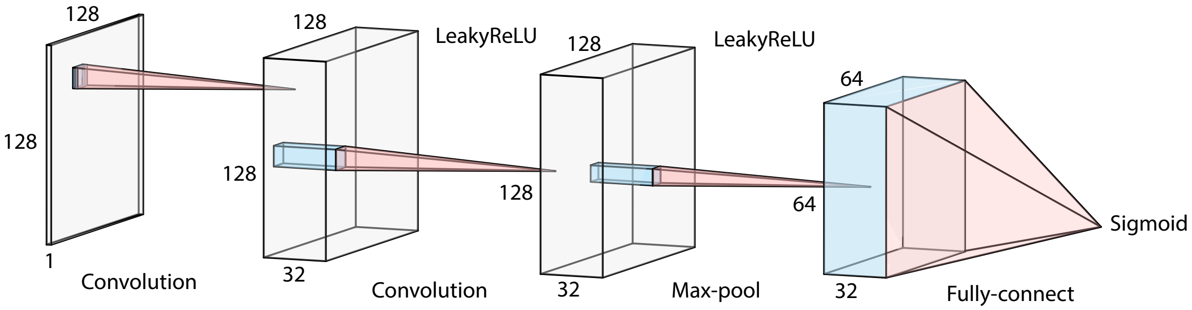


Figure 3.2: One instance of the CNN architecture employed for approximating the IO test statistic. (© IEEE 2019)

instance of the implemented CNN architecture is illustrated in Fig. 3.2. These CNNs were trained by minimizing the error function defined in Eq. (3.4) on mini-batches at each iteration. Each mini-batch contained 200 signal-absent images and 200 signal-present images. Because the HO detection performance is a lower bound of the IO detection performance, the selected optimal CNN should not perform worse than the SLNN-approximated HO (SLNN-HO) on the corresponding signal detection task if that CNN approximates IO. If this occurs, the architecture parameters need to be re-specified and a different family of CNN architectures should be considered.

The Adam algorithm [63], which is a stochastic gradient descent algorithm, was employed in Tensorflow [1] to minimize the error functions for approximating the IO and HO. All networks were trained on a single NVIDIA TITAN X GPU.

3.5 Results

3.5.1 SKE and BKE signal detection task

3.5.1.1 HO approximation

A linear SLNN was trained for 1000 mini-batches and the weight vector $\mathbf{w}$ that produced the maximum SNR_t value evaluated on the validation dataset was selected to approximate the Hotelling template. The linear templates employed by the SLNN-HO and the analytical HO are shown in Fig. 3.3. The results corresponding to the SLNN-HO closely approximate those of the analytical HO.

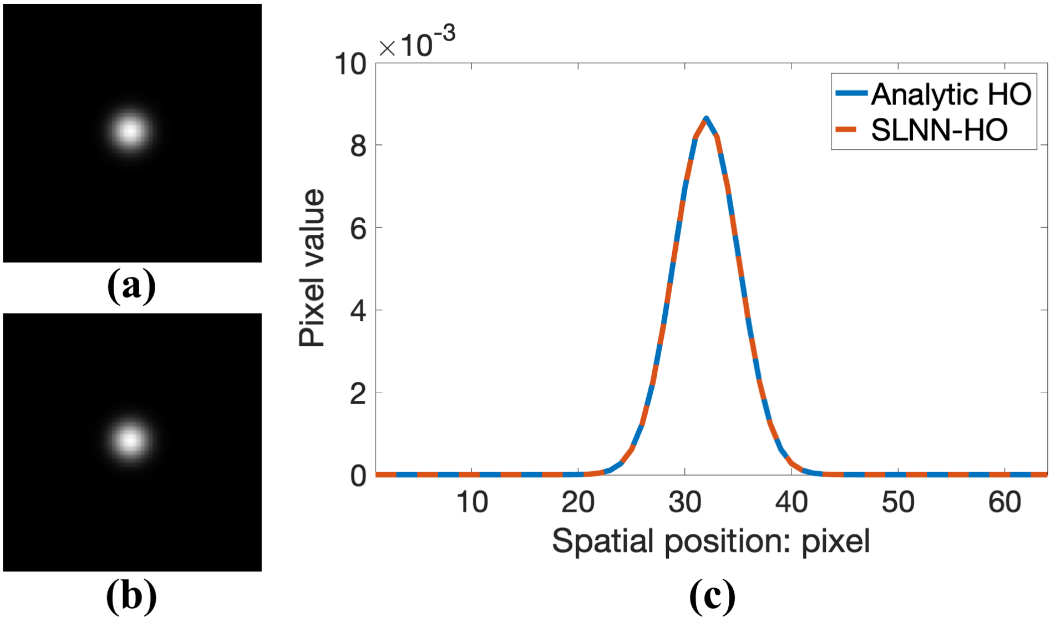
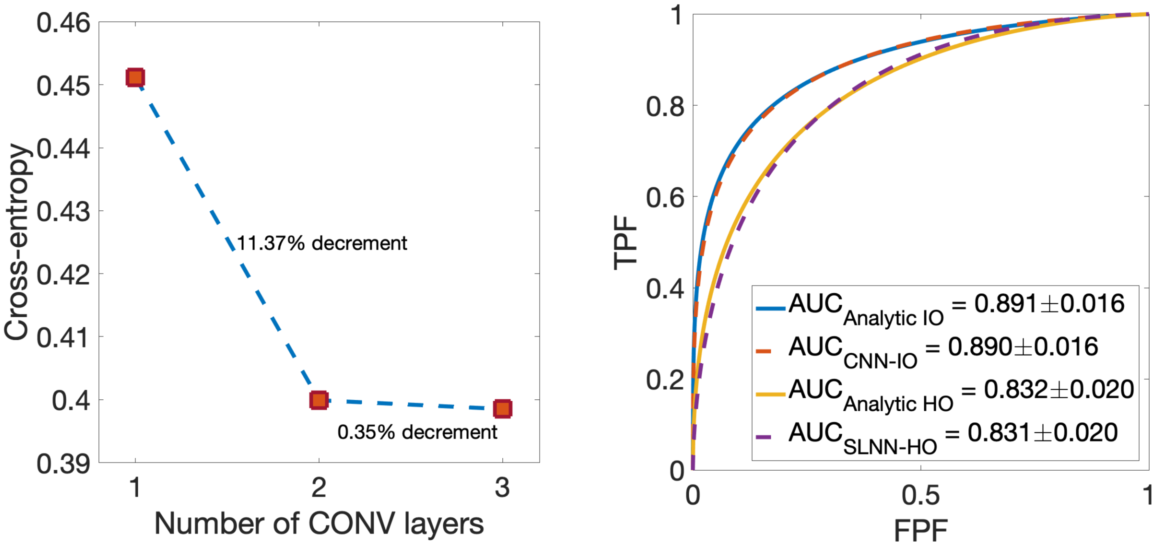


Figure 3.3: Comparison of the Hotelling template in the SKE/BKE case: (a) Analytical Hotelling template; (b) SLNN-HO template; (c) Center line profiles in (a) and (b). The estimated templates are nearly identical. (© IEEE 2019)

The ROC curve produced by the SLNN-HO (purple dashed curve) is compared to that produced by the analytical HO (yellow curve) in Fig. 3.4 (b). These two curves nearly overlap.

3.5.1.2 IO approximation

The CNNs having one to three CONV layers were trained for 100,000 mini-batches and the corresponding validation cross-entropy values are plotted in Fig. 3.4 (a). The validation cross-entropy was not significantly decreased after adding the third CONV layer. Therefore, we stopped adding more CONV layers and the CNN having the minimum validation cross-entropy, which was the CNN that possesses 3 CONV layers, was selected. The detection performance of this selected CNN was evaluated on the testing dataset and the resulting AUC value was 0.890, which was greater than that of the SLNN-HO (i.e., 0.831). Subsequently, the selected CNN was employed to approximate the IO. The testing ROC curve of the CNN-approximated IO (CNN-IO) (red-dashed curve) was compared to that of the analytical IO (blue curve) in Fig. 3.4 (b). The efficiency of the CNN-IO, which can be computed as the squared ratio of the detectability index [83] of the CNN-IO to that of the IO, was 99.14%. The mean squared error (MSE) of the posterior probabilities computed by the analytical IO and the CNN-IO was 0.30%. These quantities were evaluated on the testing dataset.



(a) (b)

Figure 3.4: (a) Validation cross-entropy values of CNNs having one to three CONV layers; (b) Testing ROC curves for the IO and HO approximations. (© IEEE 2019)

3.5.2 SKE and BKS signal detection task with a lumpy background

3.5.2.1 HO approximation

The SLNN was trained for 1000 mini-batches (i.e., 2 epochs) and the weight vector **w** that produced the maximum SNR_t value evaluated on the validation dataset was selected to approximate the Hotelling template. The linear templates employed by the SLNN-HO and the traditional HO are shown in Fig. 3.5. The results corresponding to the SLNN-HO closely approximate those of the traditional HO.

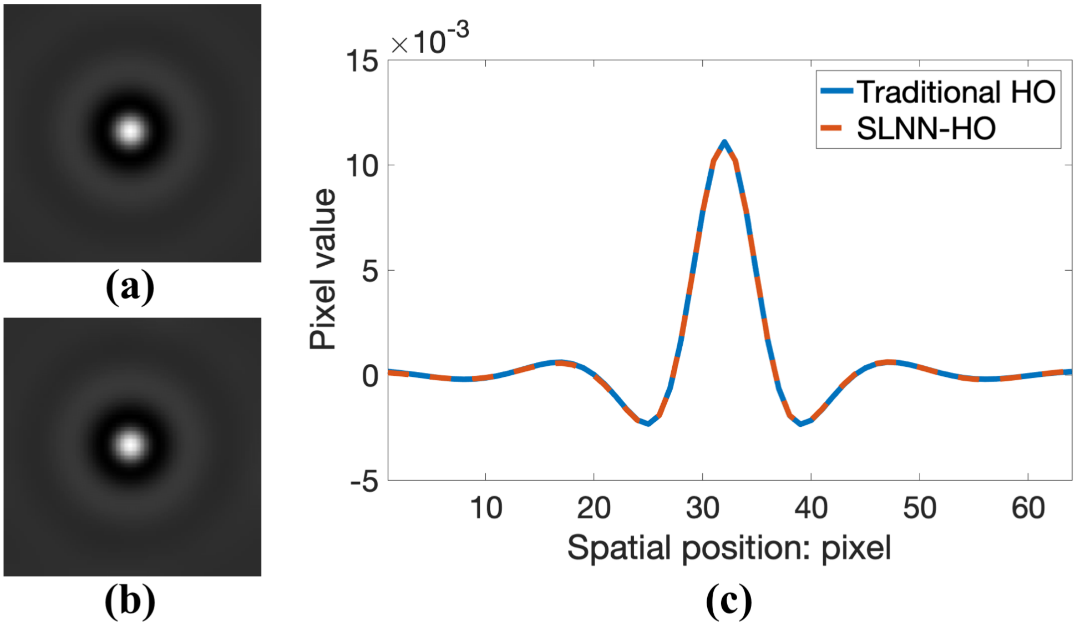
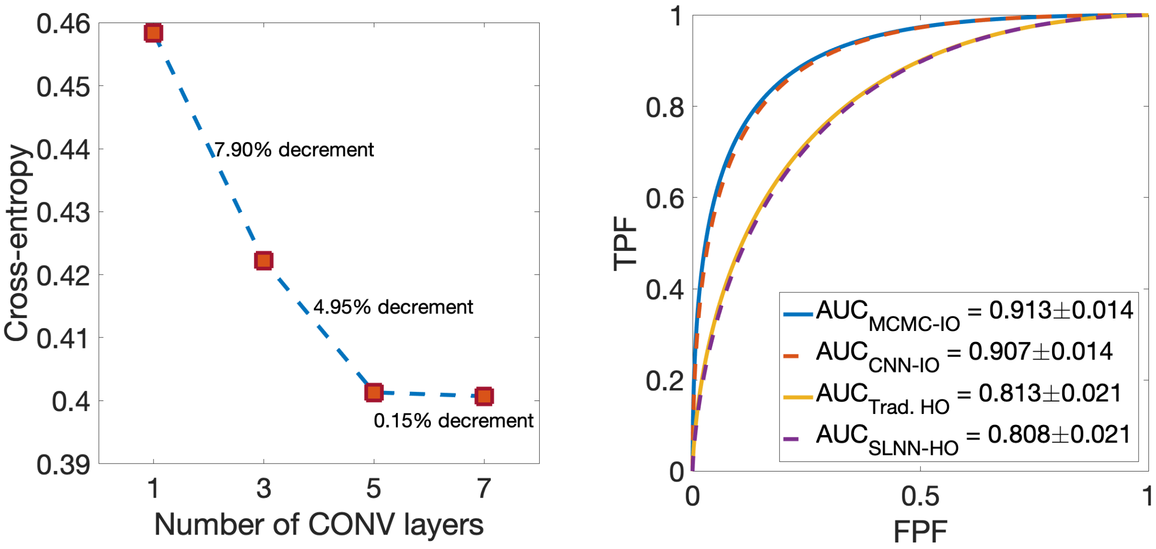


Figure 3.5: Comparison of the Hotelling template in the SKE/BKS case: (a) Traditional Hotelling template; (b) SLNN-HO template; (c) Center line profiles in (a) and (b). The estimated templates are nearly identical. (© IEEE 2019)

The ROC curves corresponding to the traditional HO (yellow curve) and the SLNN-HO (purple-dashed curve) are compared in Fig. 3.6 (b). Two ROC curves nearly overlap.

3.5.2.2 IO approximation

The CNNs having 1, 3, 5, and 7 CONV layers were trained for 100,000 mini-batches (i.e., 200 epochs) and the corresponding validation cross-entropy values are plotted in Fig. 3.6 (a). There was no significant difference of the validation cross-entropy between the CNNs having 5 and 7 CONV layers. Therefore, we stopped adding more CONV layers and the CNN having the minimum validation cross-entropy, which was the CNN that possesses 7 CONV layers, was selected. The selected CNN was evaluated on the testing dataset and the resulting AUC value was 0.907, which was greater than that of the SLNN-HO (i.e., 0.808). Subsequently, the selected CNN was employed to approximate the IO. The testing ROC curve of the CNN-IO (red-dashed curve) is compared to that of the MCMC-computed IO (MCMC-IO) (blue curve) in Fig. 3.6 (b). The efficiency of the CNN-IO was 94.64% with respect to the MCMC-IO, and the MSE of the posterior probabilities computed by the CNN-IO and the MCMC-IO was 0.84%. These quantities were evaluated on the testing dataset.



(a) (b)

Figure 3.6: (a) Validation cross-entropy values of CNNs having one to seven CONV layers; (b) Testing ROC curves for the IO and HO approximations. (© IEEE 2019)

3.5.3 SKS and BKS signal detection task with a lumpy background

3.5.3.1 HO approximation

A linear SLNN was trained for 1000 mini-batches (i.e., 2 epochs) and the weight vector $\mathbf{w}$ that produced the maximum SNR_t value evaluated on the validation dataset was selected to approximate the Hotelling template. The linear templates employed by the SLNN-HO and the traditional HO are shown in Fig. 3.7. The results corresponding to the SLNN-HO closely approximate those of the traditional HO.

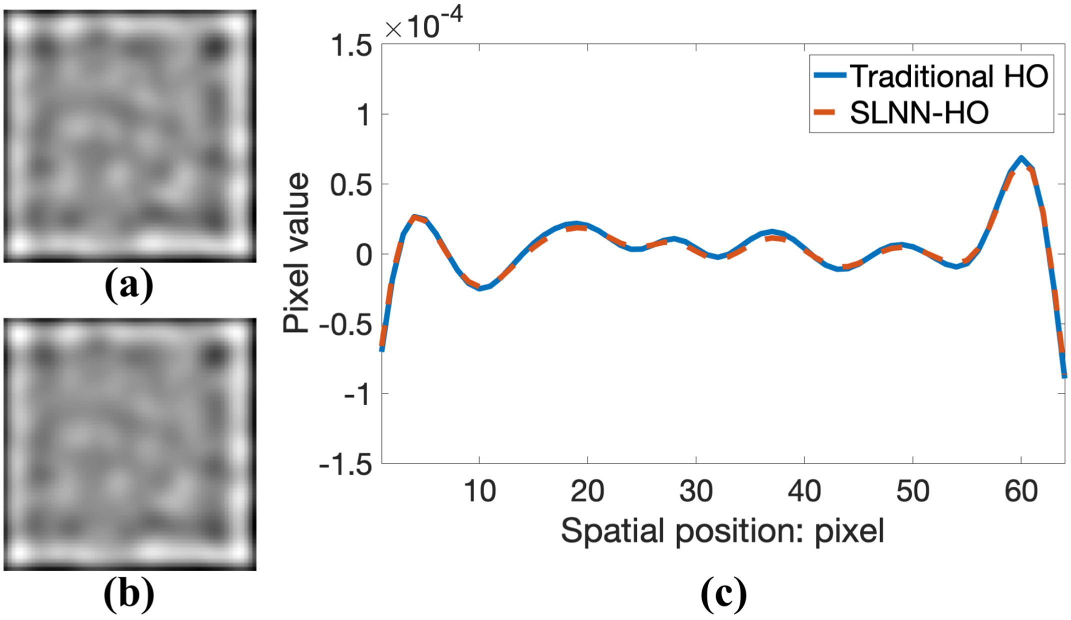
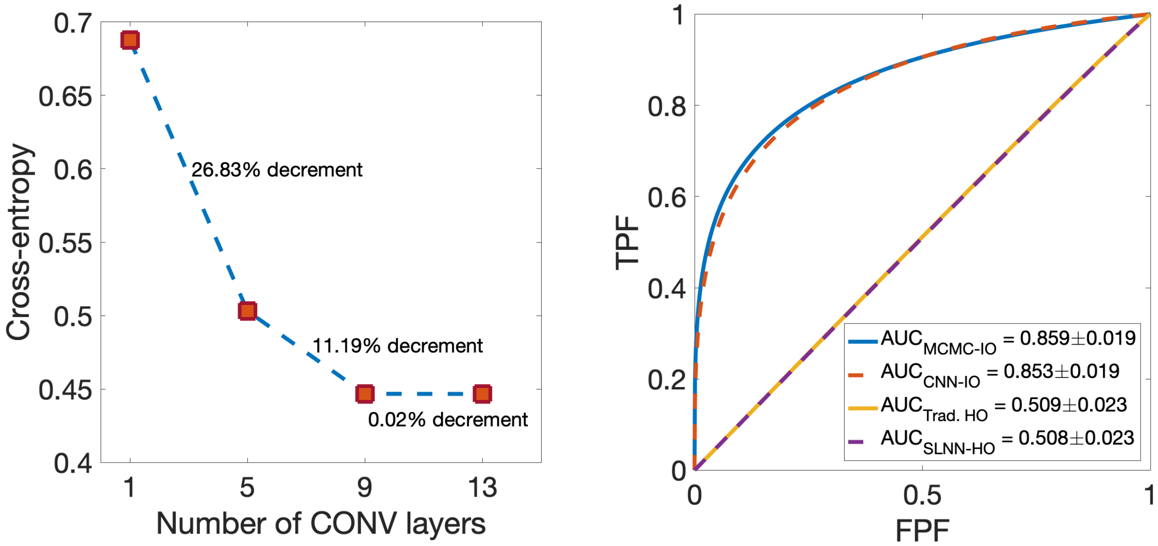


Figure 3.7: Comparison of the Hotelling template in the SKS/BKS case: (a) Traditional Hotelling template; (b) SLNN-HO template; (c) Center line profiles in (a) and (b). The estimated templates are nearly identical. (© IEEE 2019)

The ROC curves corresponding to the SLNN-HO (purple dashed curve) and the traditional HO (yellow curve) are compared in Fig. 3.8 (b). The two ROC curves nearly overlap. The HO performed nearly as a random guess for this task as expected.

3.5.3.2 IO approximation

Convolutional neural networks having 1, 5, 9, and 13 CONV layers were trained for 300,000 mini-batches (i.e., 600 epochs) and the corresponding validation cross-entropy values are plotted in Fig. 3.8 (a). Because there was no significant decrement of the validation cross-entropy value after adding 4 CONV layers to the CNN having 9 CONV layers, we stopped adding more CONV layers and the CNN having the minimum validation cross-entropy value, which was the CNN with 13 CONV layers, was selected. The selected CNN was evaluated on the testing dataset and the resulting AUC value was 0.853, which was greater than that of the SLNN-HO (i.e., 0.508). Subsequently, the selected CNN was employed to approximate the IO. The testing ROC curve produced by the CNN-IO (red-dashed curve) is compared to that produced by the MCMC-IO (blue curve) in Fig. 3.8 (b). The efficiency of the CNN-IO was 95.14% with respect to the MCMC-IO and the MSE of the posterior probabilities computed by the CNN-IO and the MCMC-IO was 1.46%. These quantities were evaluated on the testing dataset.

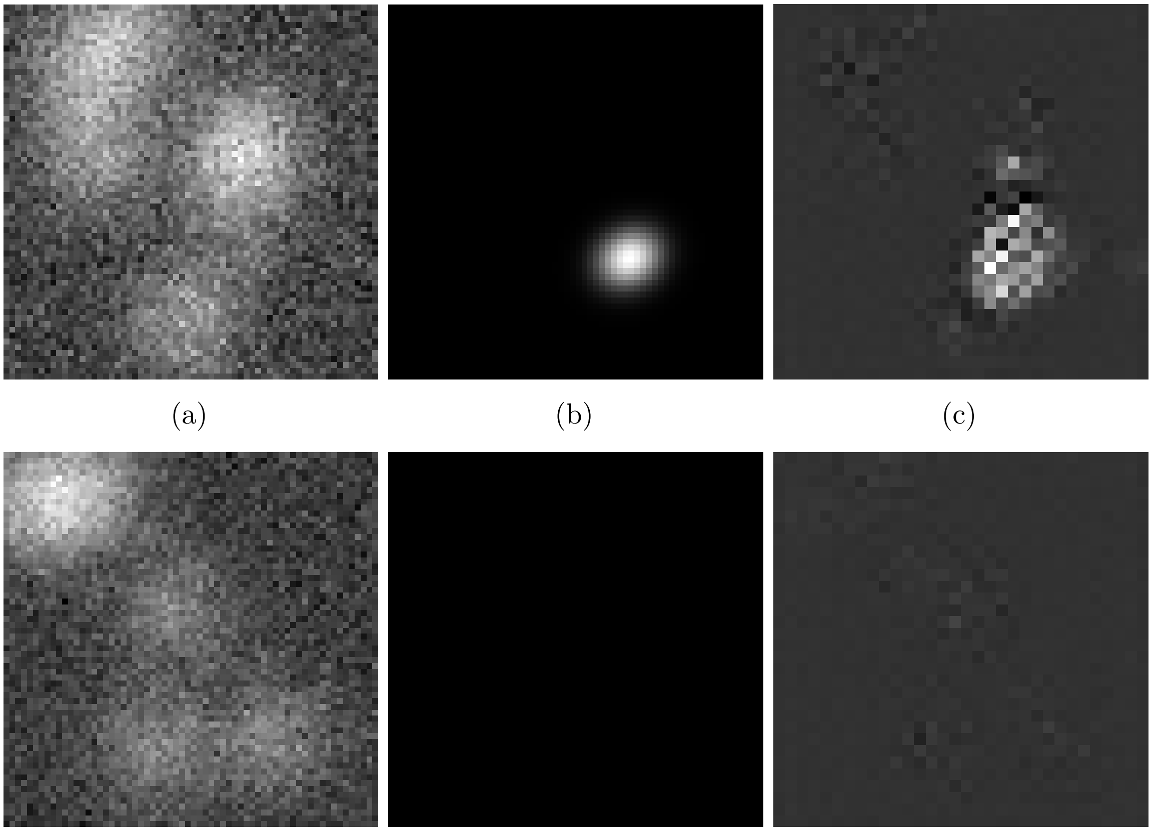


(a) (b)

Figure 3.8: (a) Validation cross-entropy values produced by CNNs having 1 to 13 CONV layers; (b) Testing ROC curves for the IO and HO approximations. (© IEEE 2019)

3.5.3.3 CNN visualization

Feature maps extracted by CONV layers enabled us to understand how CNNs were able to extract task-specific features for performing signal detection tasks. In this case, the 32 subsampled feature maps output from the max-pooling layer were weighted by the weight parameters of the last FC layer and then summed to produce a single 2D image for the visualization. That single 2D image was referred to as the signal feature map and is shown in Fig. 3.9. The signal to be detected was nearly invisible in the signal-present measurements but can be easily observed in the signal feature map.



(a) (b) (c)

(d) (e) (f)

Figure 3.9: (a) Signal-present measurements; (b) Image showing the signal contained in (a); (c) The signal feature map corresponding to (a); (d) Signal-absent measurements; (e) Image showing that the signal is absent in (d); (f) The signal feature map corresponding to (d). In the signal feature maps, the regions around the signals were activated by the CNN. (© IEEE 2019)

3.5.4 SKE and BKS signal detection task with a CLB

3.5.4.1 HO approximation

The SLNN was trained for 40,000 mini-batches (i.e., 20 epochs) and the weight vector $\mathbf{w}$ that produced the maximum validation SNR_t was selected to approximate the Hotelling template.

The traditional HO template and the SLNN-HO template are compared in Fig. 3.10. The results corresponding to the SLNN-HO closely approximate those of the traditional HO.

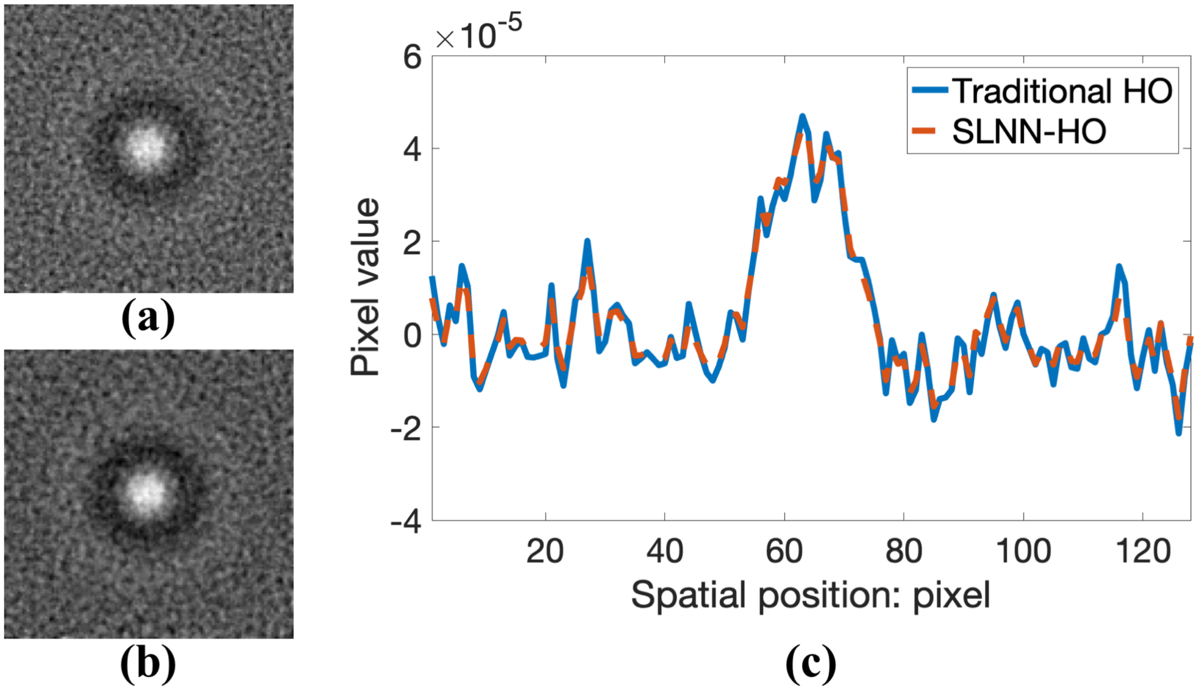
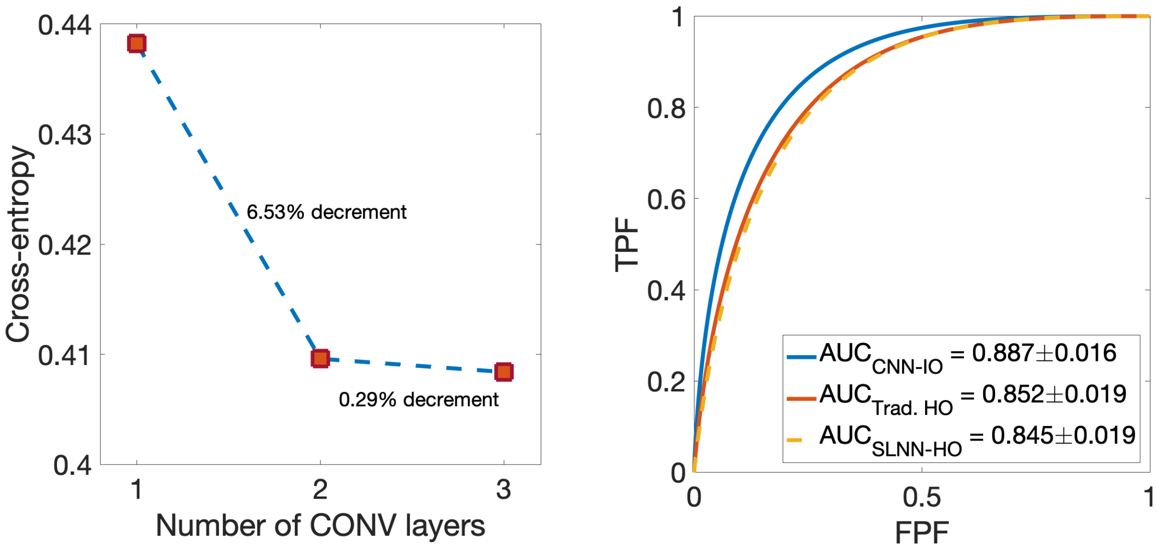


Figure 3.10: Comparison of the Hotelling template: (a) Traditional Hotelling template; (b) SLNN-HO template; (c) Center line profiles in (a) and (b). The estimated templates are nearly identical. (© IEEE 2019)

The ROC curve of the SLNN-HO (yellow-dashed curve) compares to that of the traditional HO (red curve) in Fig. 3.11 (b). Two curves nearly overlap.



(a) (b)

Figure 3.11: (a) Validation cross-entropy values of CNNs having one to three CONV layers; (b) Testing ROC curves for the IO and HO approximations. (© IEEE 2019)

3.5.4.2 IO approximation

Convolutional neural networks having one to three CONV layers were trained for 100,000 mini-batches (i.e., 50 epochs) and the corresponding validation cross-entropy values are plotted in Fig. 3.11 (a). Because the validation cross-entropy was not significantly decreased by adding the third CONV layer, we stopped adding more CONV layers and the CNN having the minimum validation cross-entropy value, which was the CNN with three CONV layers, was selected. The detection performance of this selected CNN was evaluated on the testing dataset and the resulting AUC value was 0.887, which was greater than that of the SLNN-HO (i.e., 0.845). Subsequently, the selected CNN was employed to approximate the IO. The CNN-IO was evaluated on the testing dataset and the resulting ROC curve is plotted in Fig. 3.11 (b). To show how the signal detection performance varied when the number of

CONV layers was increased, the AUC values evaluated on the testing dataset corresponding to the CNNs with one to three CONV layers are illustrated in Fig. 3.12. These AUC values were estimated by use of the "proper" binormal model [77, 87]. The AUC value was increased when more CONV layers were employed until convergence.

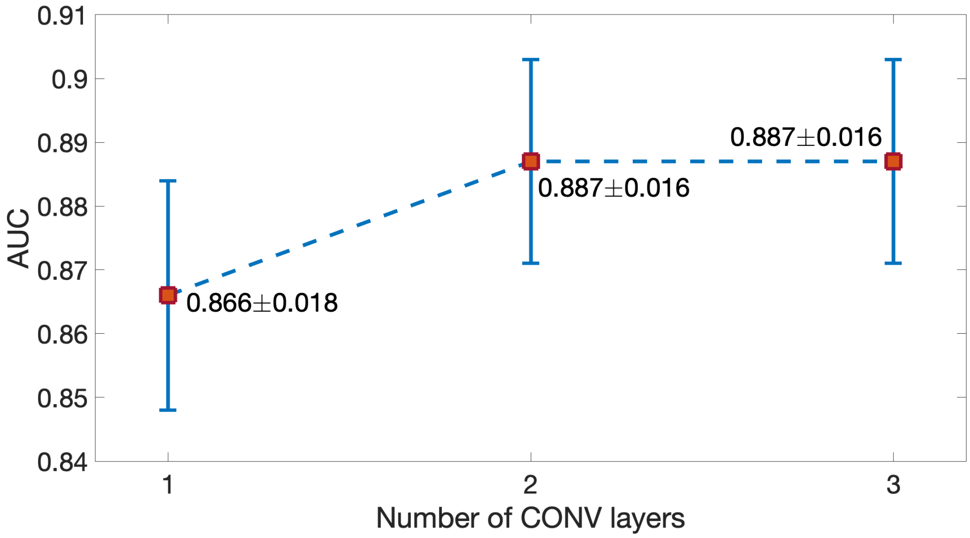
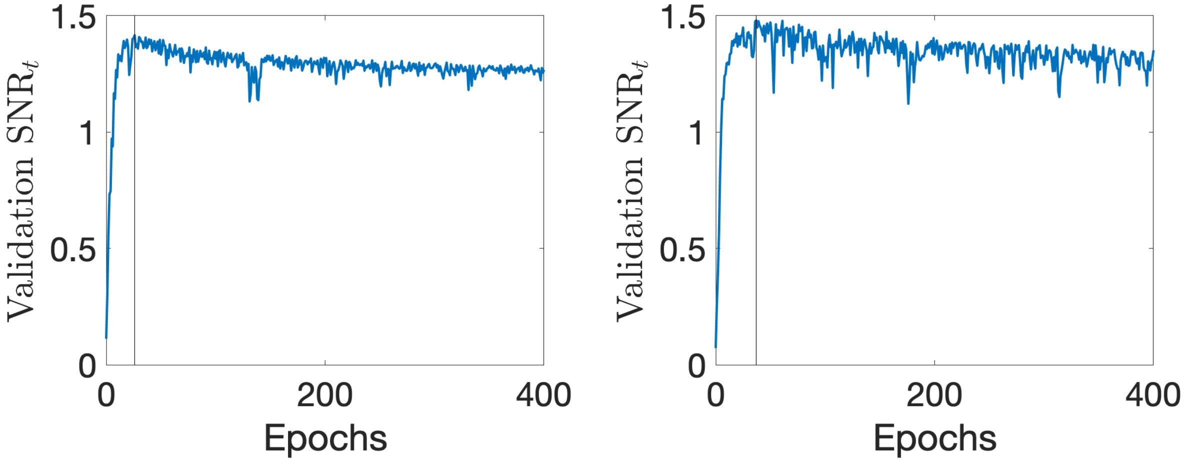


Figure 3.12: Testing AUC values of CNNs having one to three CONV layers. (© IEEE 2019)

Because MCMC applications to the CLB object model have not been reported to date, validation for the IO approximation was not provided in this case. To the best of our knowledge, we are the first to approximate the IO test statistic for the CLB object model.

3.5.4.3 HO approximation from a reduced number of images

To solve the dimensionality problem of inverting a large covariance matrix for computing the Hotelling template, the matrix-inversion lemma has been implemented in which the covariance matrix is approximated by use of a small number of images [9]. However, this method can introduce significant positive bias on the estimate of SNR_{HO} [65]. To investigate the ability of our proposed methods to approximate the HO performance when small dataset is employed, the linear SLNNs were trained by minimizing Eq. (3.11) and Eq. (3.15) on 2000 noisy measurements and 2000 background images, respectively, for 400 epochs. In the training processes, overfitting occurred as revealed by the curves of validation SNR_t with respect to the number of epochs shown in Fig. 3.13.



(a) (b)

Figure 3.13: Curves of validation SNR_t with respect to the number of epochs. (a) Validation SNR_t curve of the SLNN trained on labeled noisy measurements. (b) Validation SNR_t curve of the SLNN trained on background images using decomposition of covariance matrix. The vertical gray line indicates the epoch having the maximum validation SNR_t value. Overfitting occurred after the overall curves of validation SNR_t start to decrease. (© IEEE 2019)

However, an early-stopping strategy can be employed in which training is stopped at the epoch having the maximum validation SNR_t. The values of SNR^2_{HO}, which were computed according to Eq. (2.21), evaluated at the 400^{th} epoch and at the epoch having the maximum validation SNR_t are shown in Table 3.2. These data reveal that overfitting caused a significant positive bias on SNR^2_{HO} while the early-stopping strategy accurately approximated the reference SNR^2_{HO}, which was computed by using the Hotelling template of the traditional HO that was shown in Fig. 3.10 (a). The Hotelling template was also computed by using the matrix-inversion lemma [9] on 2000 background images, and the corresponding SNR_{HO}^2 had a significant positive bias shown in Table 3.2 as observed by others [65].

Table 3.2: SNR_{HO}^2 computed from both background images $\mathbf{b}$ and measurements $\mathbf{g}$. The Hotelling template computed from few images can cause significant positive bias. However, when SLNNs were trained using our proposed methods, early-stopping strategy in which the epoch having the maximum validation SNR_t was selected could be employed to closely approximate the HO performance. (© IEEE 2019)

Methods	400^{th} epoch	Early-stopping
Minimizing Eq. (3.11)	4.0421	**2.0940**
Minimizing Eq. (3.15)	3.1101	**2.1380**
Matrix-inversion lemma	5.7979	
Reference	**2.1075**	

3.6 Discussion and Conclusion

In this chapter, we proposed a supervised learning-based method that employs CNNs to approximate the IO for binary signal detection tasks. This method represents an alternative approach to conventional numerical approaches such as MCMC methods for estimating the IO performance to be used in optimizing medical imaging systems and data-acquisition designs. Practical advantages of the proposed method over the MCMC methods exist. To run MCMC methods appropriately, practical issues such as designs of proposal density functions from which Markov chains can be effectively generated need to be addressed. Current applications of the MCMC methods have been limited to some specific object models that include parameterized torso phantoms [50], lumpy background models [68] and a binary texture model [2]. Supervised learning-based approaches may be easier to deploy with sophisticated object models than are MCMC methods. To demonstrate this, in the numerical study, we applied the proposed supervised learning method with a CLB object model, for which the IO computation has not been addressed by MCMC methods to date [2]. Another practical advantage of the proposed method is that supervised learning-based methods are becoming widespread in their usage and many researchers are becoming experienced on training feed-forward ANNs.

In this study, to specify the CNN architecture for approximating the IO, we explored a family of CNNs that possess different numbers of CONV layers. By adding more CONV layers, the representation capacity of the network is increased and the test statistic can be more accurately approximated. This study does not investigate other architecture parameters such as the number of FC layers and the size of convolutional filters. Recent work [31] proposed a method that optimizes the network architecture in the training process. This represents a possible approach for jointly optimizing the network architecture and weights to approximate the IO test statistic.

We also proposed a supervised learning-based method using a simple linear SLNN to approximate the HO that represents the optimal linear observer. The proposed methodology directly learns the Hotelling template without estimating and inverting covariance matrices. Accordingly, the proposed method can scale well to large images. When approximating the HO test statistic, one can employ a linear SLNN because the HO test statistic depends linearly on the input image. We also provided an alternative method to learn the HO by use of a covariance-matrix decomposition. The feasibility of both methods to learn the HO from a reduced number of images was investigated. For the case where 2000 clustered lumpy images with the dimension 128×128 were employed to approximate the HO, our proposed learning-based methods could still produce accurate estimates of SNR_{HO} by incorporating an early-stopping strategy.

Numerous topics remain for future investigation. With regards to approximating IOs by use of experimental images, there is a need to investigate methods to train CNN models by use of limited training data. To accomplish this, one may investigate transfer learning [88] or domain adaptation methods [39] to learn features of images in target domain (e.g., experimental images) by use of images in source domain (e.g., computer-simulated images). Given experimental images, one may also employ the method proposed by Kupinski *et al.* [66] or train a generative adversarial network [46] to establish a stochastic object model (SOM) that can produce large datasets for use in the CNN training. Finally, it will be important to extend the proposed methods to more complicated tasks. A supervised learning method to approximate the IO for joint signal detection and localization tasks is provided in the next chapter.

Chapter 4

Approximating the Ideal Observer for joint signal detection and localization tasks by use of supervised learning method

4.1 Overview

Joint signal detection and localization (detection-localization) tasks are often considered in medical imaging [42, 44, 100, 108, 116]. When imaging systems and data-acquisition designs are optimized for such tasks, the Ideal Observer (IO) that maximizes the area under the LROC curve (ALROC) has been advocated to provide a FOM. As discussed in Chapter 2, the IO implements a modified generalized likelihood ratio test (MGLRT) [62]. However, the MGLRT employs likelihood ratios that are generally intractable to compute analytically. To address this limitation, sampling-based methods that employ Markov-Chain Monte Carlo (MCMC) techniques [68] have been developed. However, to run MCMC methods appropriately, practical issues such as the design of proposal density function from which Markov chains can be simulated in a efficient way need to be addressed. Current applications of MCMC methods have been limited to some specific object models that include lumpy object models [68], binary texture models [2] and parameterized torso phantoms [50]. It remains unclear how MCMC methods can be applied for cases where the objects to-be-imaged cannot be described by these object models.

Computer-simulation is an important approach that is commonly employed for the design and optimization of imaging systems. In such cases, supervised learning methods can be implemented with large amounts of simulated data to train inference models that are represented by artificial neural networks (ANNs) for establishing numerical observers. A previous work of Kupinski *et al.* explored the use of fully-connected neural networks (FCNNs) to approximate the IO that acts on low-dimensional feature vectors for binary signal detection tasks [67]. In Chapter 3, we investigated a supervised learning-based method that employs convolutional neural networks (CNNs) to approximate the IO that acts on 2D image data for binary signal detection tasks.

In this chapter, a supervised learning-based method that employs CNNs to approximate the IO for signal detection-localization tasks is explored. The proposed method represents a deep-learning-based implementation of the IO decision strategy proposed in the seminal theoretical work by Khurd and Gindi [62]. The considered signal detection-localization tasks involve various object models in combination with several realistic measurement noise models. Numerical observer performance is assessed via the LROC analysis. The results of the proposed supervised-learning method are compared to those produced by the MCMC method or analytical computations when feasible.

4.2 Approximating the IO for signal detection-localization tasks by use of CNNs

As discussed in Chapter 2, the IO for signal detection-localization tasks implements a MGLRT. According to Eq. (2.26), the MGLRT is equivalent to a posterior ratio test. Therefore, to approximate the IO for a signal detection-localization task, a CNN can be trained to approximate a set of posterior probabilities that are employed in the posterior ratio test in Eq. (2.26). To achieve this, the softmax function is employed in the last layer of the CNN, the so-called softmax layer [90], so that the output of the CNN can be interpreted as probabilities. Let $\mathbf{\Theta}$ denote the vector of weight parameters of a CNN and let $\mathbf{z}(\mathbf{g};\mathbf{\Theta}) \in \mathbb{R}^{J+1}$ denote the output of the last hidden layer of the CNN, which is also the input to the softmax layer. The CNN-approximated posterior probabilities can be computed as:

$$\Pr(H_j|\mathbf{g},\mathbf{\Theta}) \equiv \frac{\exp[z_j(\mathbf{g};\mathbf{\Theta})]}{\sum_{j'=0}^{J}\exp[z_{j'}(\mathbf{g};\mathbf{\Theta})]},\ j=0,1,...,J, \tag{4.1}$$

where $z_j(\mathbf{g};\mathbf{\Theta})$ is the $(j+1)^{th}$ element of $\mathbf{z}(\mathbf{g};\mathbf{\Theta})$. The CNN parameter vector $\mathbf{\Theta}$ is to be determined such that the difference between the CNN-approximated posterior probability $\Pr(H_j|\mathbf{g},\mathbf{\Theta})$ and the actual posterior probability $\Pr(H_j|\mathbf{g})$ is minimized.

The maximum likelihood (ML) estimate of $\mathbf{\Theta}$ can be approximated by use of a supervised learning method [67]. Let $y \in \{0,1,...,J\}$ denote the label of the measured image $\mathbf{g}$, where $y=j$ corresponds to the hypothesis H_j. Given the joint probability distribution $p(\mathbf{g},H_y)$, the ML estimate of the CNN weight parameters $\mathbf{\Theta}_{ML}$ can be obtained by minimizing the generalization error, which is defined as the ensemble average of the cross-entropy over the distribution $p(\mathbf{g}, H_y)$ [67, 123]:

$$\mathbf{\Theta}_{ML} = \arg\min_{\mathbf{\Theta}} \langle -\log[\Pr(H_y|\mathbf{g}, \mathbf{\Theta})] \rangle_{(\mathbf{g},y)}, \tag{4.2}$$

where $\langle \cdot \rangle_{(\mathbf{g},y)}$ denotes the ensemble average over the distribution $p(\mathbf{g}, H_y)$. When the CNN possesses sufficient representation capacity such that $\mathbf{z}(\mathbf{g}; \mathbf{\Theta})$ can take any functional form, $\Pr(H_j|\mathbf{g}, \mathbf{\Theta}_{ML}) = \Pr(H_j|\mathbf{g})$. To see this, one can compute the gradient of the cross-entropy with respect to $z_j(\mathbf{g}; \mathbf{\Theta})$ as:

$$\begin{aligned} &\frac{\partial \langle -\log[\Pr(H_y|\mathbf{g}, \mathbf{\Theta})] \rangle_{(\mathbf{g},y)}}{\partial z_j(\mathbf{g}; \mathbf{\Theta})} \\ &= p(\mathbf{g}) \Big[\frac{\exp[z_j(\mathbf{g}; \mathbf{\Theta})]}{\sum_{j'=0}^{J} \exp[z_{j'}(\mathbf{g}; \mathbf{\Theta})]} - \Pr(H_j|\mathbf{g}) \Big]. \end{aligned} \tag{4.3}$$

The derivation of this gradient computation can be found in Appendix A. Because $\mathbf{z}(\mathbf{g}; \mathbf{\Theta})$ can take any functional form when the CNN possesses sufficient representation capacity, determining $\mathbf{\Theta}_{ML}$ involves finding $\mathbf{z}(\mathbf{g}; \mathbf{\Theta})$ that minimizes the cross-entropy defined in Eq. (4.2). According to Eq. (4.3), for any $\mathbf{g} \in \{\mathbf{g}|p(\mathbf{g}) \neq 0\}$, the optimal solution $z_j(\mathbf{g}; \mathbf{\Theta}_{ML})$ that has zero gradient value satisfies $\frac{\exp[z_j(\mathbf{g};\mathbf{\Theta}_{ML})]}{\sum_{j'=0}^{J} \exp[z_{j'}(\mathbf{g};\mathbf{\Theta}_{ML})]} = \Pr(H_j|\mathbf{g})$, from which $\Pr(H_j|\mathbf{g}, \mathbf{\Theta}_{ML}) = \Pr(H_j|\mathbf{g})$.

Given a training dataset that contains N independent training samples $\{(\mathbf{g}_i, y_i)\}_{i=1}^{N}$, $\mathbf{\Theta}_{ML}$ can be estimated by minimizing the empirical error as:

$$\hat{\mathbf{\Theta}}_{ML} = \arg\min_{\mathbf{\Theta}} \frac{1}{N} \sum_{i=1}^{N} -\log[\Pr(H_{y_i}|\mathbf{g}_i, \mathbf{\Theta})], \tag{4.4}$$

where $\hat{\mathbf{\Theta}}_{ML}$ is an empirical estimate of $\mathbf{\Theta}_{ML}$. The posterior probability $\Pr(H_j|\mathbf{g})$ can be subsequently approximated by the CNN-represented posterior probability $\Pr(H_j|\mathbf{g}, \hat{\mathbf{\Theta}}_{ML})$ and the decision strategy described in Eq. (2.26) can be implemented. It should be noted that minimizing empirical error on a small training dataset can result in overfitting and large generalization errors. Mini-batch stochastic gradient descent algorithms can be employed to reduce the rate of overfitting [45]. These mini-batches can be generated on-the-fly when online learning is implemented.

4.3 Numerical studies

Computer-simulation studies were conducted to investigate the supervised learning-based method for approximating the IO for signal detection-localization tasks. The considered signal detection-localization tasks included two background-known-exactly (BKE) tasks and two background-known-statistically (BKS) tasks. A lumpy background (LB) model [64] and a clustered lumpy background (CLB) model [13] were employed in the BKS tasks.

The imaging system considered was an idealized parallel-hole collimator system that was described by a linear C-D mapping with Gaussian point response functions (PRFs) given by [66, 68]:

$$h_m(\mathbf{r}) = \frac{\hbar}{2\pi w_h^2} \exp\left(-\frac{(\mathbf{r}-\mathbf{r}_m)^T(\mathbf{r}-\mathbf{r}_m)}{2w_h^2}\right), \tag{4.5}$$

where $\hbar$ and w_h are the height and width of the PRFs, respectively. Imaging systems with larger $\hbar$ have greater sensitivity while imaging systems with larger w_h have lower resolution.

Denote $\tilde{\mathbf{r}}_j = \mathbf{R}_{\theta_j}(\mathbf{r}-\mathbf{r}_{c_j})$, the signal to be detected and localized was modeled by a 2D Gaussian function with 9 possible locations:

$$f_{s_j}(\mathbf{r}) = a_{s_j} \exp\left(-\tilde{\mathbf{r}}_j^T \mathbf{D}_j^{-1} \tilde{\mathbf{r}}_j\right), \tag{4.6}$$

where a_{s_j} is the signal amplitude, $\mathbf{R}_{\theta_j} = \begin{bmatrix} \cos(\theta_j) & -\sin(\theta_j) \\ \sin(\theta_j) & \cos(\theta_j) \end{bmatrix}$ is a rotation matrix corresponding to the rotating angle θ_j, $\mathbf{D}_j = \begin{bmatrix} 2w_{1_j}^2 & 0 \\ 0 & 2w_{2_j}^2 \end{bmatrix}$ is a matrix that determines the width of the j^{th} signal along each axis, and $\mathbf{r}_{c_j}$ is the center location of the j^{th} signal. With consideration of the specified imaging system, the m^{th} element s_{j_m} of the signal image $\mathbf{s}_j$ can be subsequently computed as:

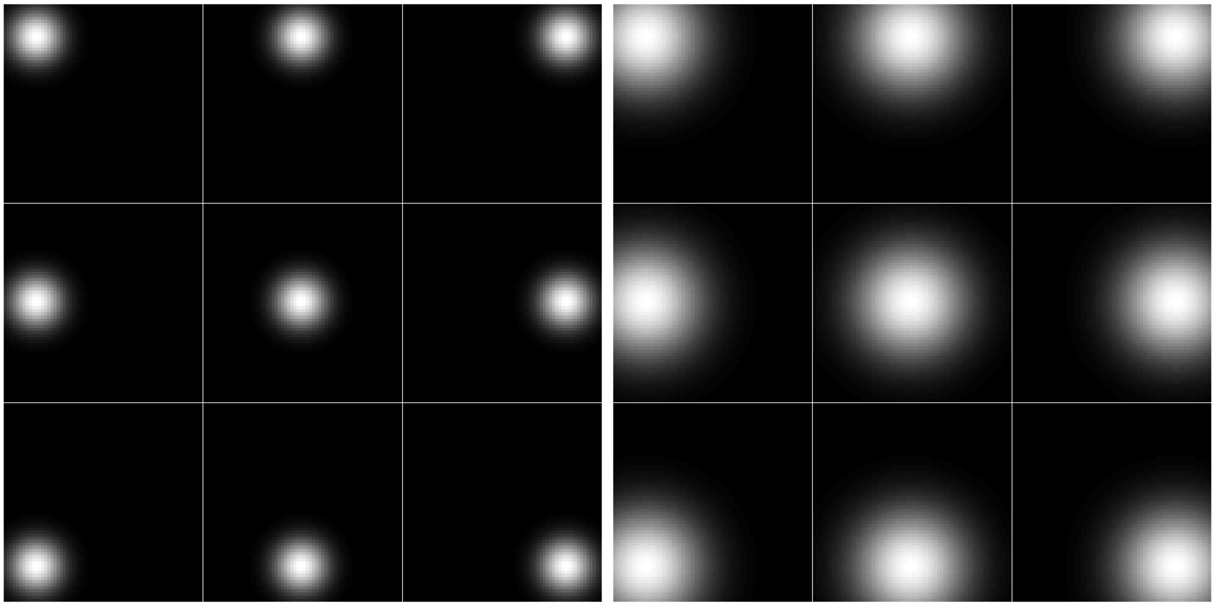
$$s_{j_m} = A_j \exp\left(-\tilde{\mathbf{r}}_{jm}^T \mathbf{D}_j'^{-1} \tilde{\mathbf{r}}_{jm}\right), \tag{4.7}$$

where $A_j = a_{s_j} \hbar w_{1_j} w_{2_j} \sqrt{\frac{1}{(w_h^2 + w_{1_j}^2)(w_h^2 + w_{2_j}^2)}}$, $\mathbf{D}_j' = \begin{bmatrix} 2(w_h^2 + w_{1_j}^2) & 0 \\ 0 & 2(w_h^2 + w_{2_j}^2) \end{bmatrix}$ and $\tilde{\mathbf{r}}_{jm} = \mathbf{R}_{\theta_j}(\mathbf{r}_m - \mathbf{r}_{c_j})$.

For each task described below, the LROC curves were fit by use of LROC software [57] that implements Swensson's fitting algorithm [101] and the IO performance was quantified by the ALROC.

4.3.1 BKE signal detection-localization tasks

For the BKE tasks, the size of background image was 64 × 64 pixels and $\mathbf{b} = \mathbf{0}$. The signal to be detected and localized had the signal amplitude $a_{s_j} = 0.2$, width $w_{1_j} = w_{2_j} = 3$, and $\mathbf{R}_{\theta_j} = 0$ for all 9 possible locations $j = 1, 2, ..., 9$. Two imaging systems described by different PRFs were considered. The first imaging system, "System 1", was described by $\hbar = 60$ and $w_h = 5$. The second imaging system, "System 2", was described by $\hbar = 144$ and $w_h = 12$. The signals at different locations imaged through the two imaging systems are illustrated in Fig. 4.1.



(a) (b)

Figure 4.1: (a) Signal images corresponding to “System 1”. (b) Signal images corresponding to “System 2”. (© IEEE 2020)

To investigate the ability of the CNN to approximate a non-linear IO test statistic, a Laplacian probability density function, which has been utilized to describe histograms of fine details in digital mammographic images [25, 52], was employed to model the likelihood function $p(\mathbf{g}|H_j)$. Specifically, the measured image data $\mathbf{g}$ were simulated by adding independent and identically distributed (i.i.d.) Laplacian noise [25]: $n_m \sim L(0,c)$, where $L(0,c)$ denotes a Laplacian distribution with the mean of 0 and the exponential decay of c, which was set to $20/\sqrt{2}$ corresponding to a standard deviation of 20. In this case, the likelihood ratio can be analytically computed as [25]:

$$\Lambda_j(\mathbf{g}) = \exp\left[\frac{1}{c}\sum_{m=1}^{M}(|g_m - b_m| - |g_m - b_m - s_{j_m}|)\right]. \tag{4.8}$$

The IO decision strategy described by Eq. (2.25) was subsequently implemented by use of the likelihood ratios given by Eq. (4.8), and the resulting LROC curves and ALROC values were compared to those produced by the proposed supervised learning method.

Rank ordering of imaging system designs depends on specifications of tasks [79]. The two imaging systems were ranked by use of the IO performance for the considered signal detection-location tasks via LROC analysis. In addition, to demonstrate that the imaging system design optimized by use of the IO for signal detection-localization tasks may differ from that optimized by use of the IO for the simplified binary signal detection tasks, the two imaging systems were also assessed by use of the IO performance for the simplified binary signal detection tasks via ROC analysis. The ROC curves were fit by use of Metz-ROC software [76] using the "proper" binormal model [77].

4.3.2 BKS signal detection-localization task with a lumpy background model

The first BKS task utilized a lumpy object model to emulate background variability [64]. The considered lumpy background models are described as [9, 64]:

$$f_b(\mathbf{r}) = \sum_{n=1}^{N_b} l(\mathbf{r} - \mathbf{r}_n | a, w_b), \tag{4.9}$$

where $N_b \sim P(\bar{N})$ denotes the number of the lumps, $P(\bar{N})$ denotes a Poisson distribution with mean of $\bar{N} = 8$, and $l(\mathbf{r} - \mathbf{r}_n | a, w_b)$ denotes the lump function that was modeled by a 2D Gaussian function:

$$l(\mathbf{r} - \mathbf{r}_n | a, w_b) = a \exp\left(-\frac{(\mathbf{r} - \mathbf{r}_n)^T(\mathbf{r} - \mathbf{r}_n)}{2w_b^2}\right). \tag{4.10}$$

Here, $a = 1$, $w_b = 7$, and $\mathbf{r}_n$ denotes the center location of the n^{th} lump that was sampled from a uniform distribution over the image field of view. The imaging system PRF was specified by $\hbar = 40$ and $w_h = 1.5$. The image size was 64×64 and the m^{th} $(1 \leq m \leq 4096)$ element of the background image b_m is given by:

$$b_m = \frac{a\hbar w_b^2}{w_h^2 + w_b^2} \exp\left(-\frac{(\mathbf{r}_m - \mathbf{r}_n)^T(\mathbf{r}_m - \mathbf{r}_n)}{2(w_h^2 + w_b^2)}\right). \tag{4.11}$$

The measurement noise considered in this case was i.i.d. Gaussian noise with a mean of 0 and a standard deviation of 20. Three realizations of the signal-absent images are shown in the top row in Fig. 4.2. The signals to be detected and localized were specified by Eq. (4.6) with $a_{s_j} = 0.5$, $w_{1_j} = w_{2_j} = 2$, and $\mathbf{R}_{\theta_j} = 0$ for all 9 possible locations $j = 1, 2, ..., 9$. The signal at different locations is illustrated in Fig. 4.3 (a).

Because the likelihood ratios $\Lambda_j(\mathbf{g})$ in this case cannot be analytically computed, the MCMC method developed by Kupinski *et al.* [68] was implemented as a reference method. The MCMC method computed the likelihood ratio as:

$$\Lambda_j(\mathbf{g}) \approx \frac{1}{N_c} \sum_{i=1}^{N_c} \Lambda_{\mathrm{BKE}_j}(\mathbf{g}|\mathbf{b}^{(i)}), \tag{4.12}$$

where $\Lambda_{\mathrm{BKE}_j}(\mathbf{g}|\mathbf{b}^{(i)}) \equiv \frac{p(\mathbf{g}|\mathbf{b}^{(i)},H_j)}{p(\mathbf{g}|\mathbf{b}^{(i)},H_0)}$ is the BKE likelihood ratio conditional on the i^{th} background image $\mathbf{b}^{(i)}$ and N_c is the number of samples used in Monte Carlo integration. Because Gaussian noise was considered in this case, $\Lambda_{\mathrm{BKE}_j}(\mathbf{g}|\mathbf{b}^{(i)})$ can be analytically computed as:

$$\Lambda_{\mathrm{BKE}_j}(\mathbf{g}|\mathbf{b}^{(i)}) = \exp\left[(\mathbf{g} - \mathbf{b}^{(i)} - \mathbf{s}_j/2)^T \mathbf{K}_n^{-1} \mathbf{s}_j\right], \tag{4.13}$$

where $\mathbf{K}_n$ is the covariance matrix of the measurement noise $\mathbf{n}$. The i^{th} background image $\mathbf{b}^{(i)}$ was sampled from the probability density function $p(\mathbf{b}|\mathbf{g}, H_0)$ by constructing a Markov chain according to the method described in [68]. Each Markov chain was simulated by running 200,000 iterations.

4.3.3 BKS signal detection-localization task with a CLB model

The second BKS task utilized a clustered lumpy background (CLB) model to emulate background variability. This model was developed to synthesize mammographic image textures [13]. In this case, the background image $\mathbf{b}$ had the dimension of 128 × 128 pixels and its m^{th} element b_m is computed as [13]:

$$b_m = \sum_{k=1}^{K} \sum_{n=1}^{N_k} l\left(\mathbf{r}_m - \mathbf{r}_k - \mathbf{r}_{kn} | \mathbf{R}_{\theta_{kn}}\right). \tag{4.14}$$

Here, K denotes the number of clusters that was sampled from a Poisson distribution with the mean of $\bar{K}$: $K \sim P(\bar{K})$, N_k denotes the number of blobs in the k^{th} cluster that was sampled from a Poisson distribution with the mean of $\bar{N}$: $N_k \sim P(\bar{N})$, $\mathbf{r}_k$ denotes the center location of the k^{th} cluster that was sampled uniformly over the image field of view, and $\mathbf{r}_{kn}$ denotes the center location of the n^{th} blob in the k^{th} cluster that was sampled from a Gaussian distribution with the center of $\mathbf{r}_k$ and standard deviation of σ. The blob function $l\left(\mathbf{r}|\mathbf{R}_{\theta_{kn}}\right)$ was specified as:

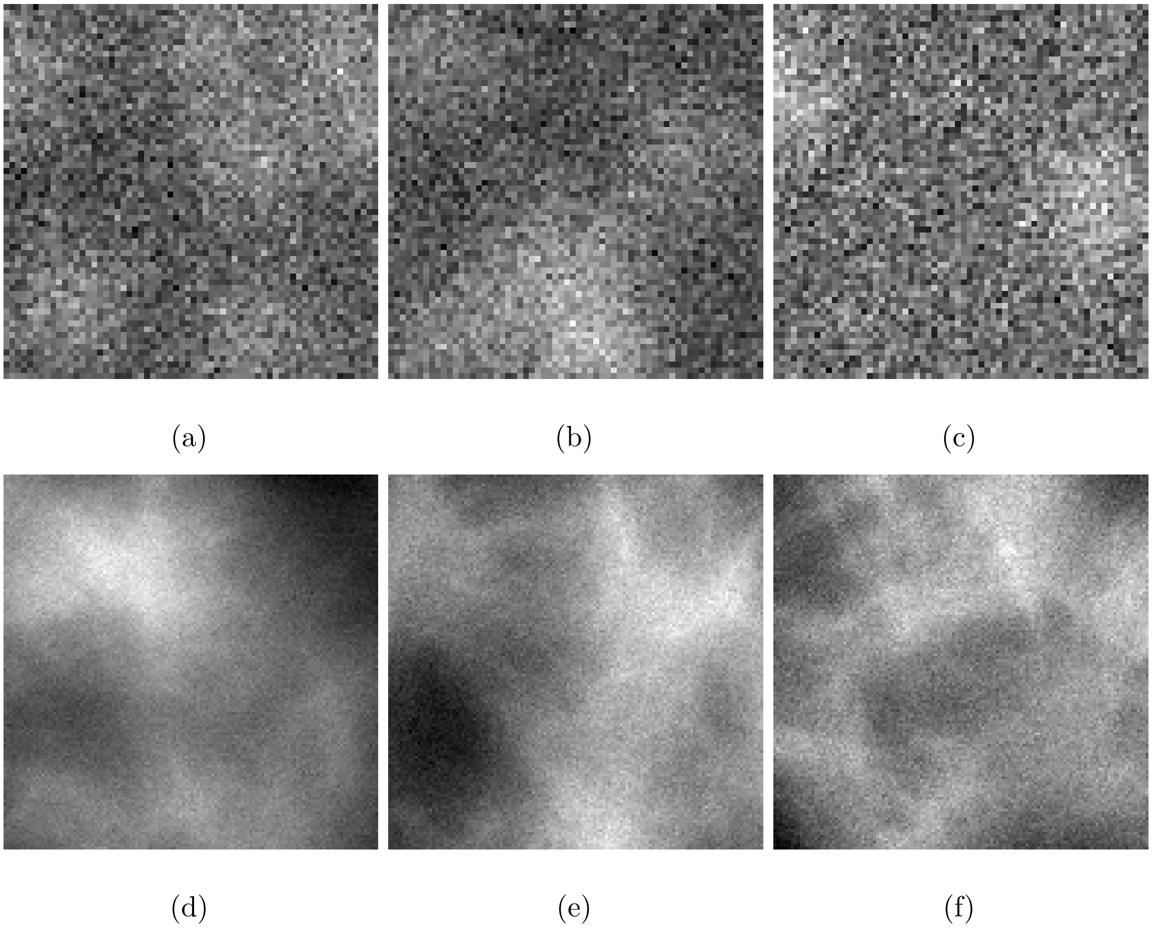
$$l\left(\mathbf{r}|\mathbf{R}_{\theta_{kn}}\right) = A \exp\left(-\alpha \frac{\|\mathbf{R}_{\theta_{kn}}\mathbf{r}\|^{\beta}}{L(\mathbf{R}_{\theta_{kn}}\mathbf{r})}\right), \tag{4.15}$$

where $L(\mathbf{r})$ is computed as the "radius" of the ellipse with half-axes L_x and L_y, and $\mathbf{R}_{\theta_{kn}}$ is the rotation matrix corresponding to the angle θ_{kn} that was sampled from a uniform distribution between 0 and 2π. The parameters of the CLB model employed in this study are shown in Table. 4.1

Table 4.1: Parameters for generating CLB images

$\overline{K}$	$\overline{N}$	L_x	L_y	α	β	σ	A
50	20	5	2	2.1	0.5	12	40

The measurement noise was modeled by a mixed Poisson-Gaussian noise model [9] in which the standard deviation of Gaussian noise was set to 20. Three examples of the signal-absent images are shown in the bottom row in Fig. 4.2.


(a) (b) (c)

(d) (e) (f)

Figure 4.2: (a)-(c) Signal-absent images corresponding to the LB model. (d)-(f) Signal-absent images corresponding to the CLB model. (© IEEE 2020)

The signal $\mathbf{s}_j$ had the amplitude of 80, the width of $\mathbf{s}_j$ along each axis took a value from {5, 8, 10}, and the rotation angle of $\mathbf{s}_j$ took a value from $\{-\pi/4, 0, \pi/4\}$. The signal at different locations is illustrated in Fig. 4.3 (b).

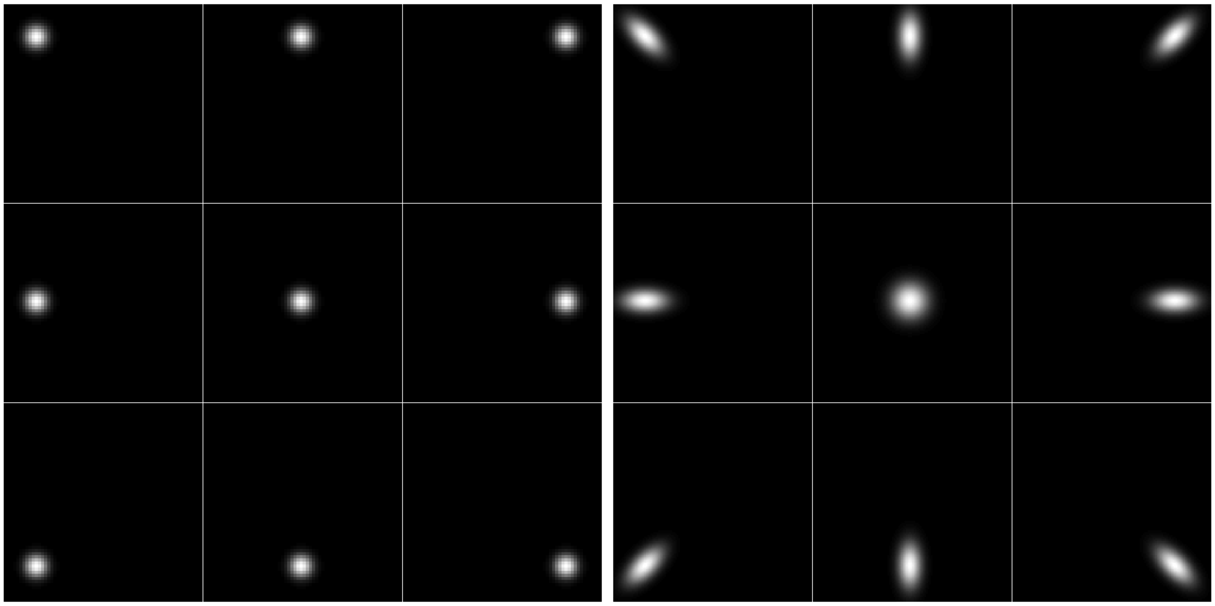

(a) (b)

Figure 4.3: (a) Signal images corresponding to the 9 possible signal locations employed in the BKS task with the LB model. (b) Signal images corresponding to the 9 possible signal locations employed in the BKS task with the CLB model. (© IEEE 2020)

4.3.4 CNN training details

The conventional train-validation-test scheme was employed to evaluate the proposed supervised learning approaches. The CNNs were trained on a training dataset, the CNN architectures and weight parameters were subsequently specified by assessing performance on a validation dataset and, finally, the performances of the CNNs on the signal detection-localization tasks were evaluated on a testing dataset. The training datasets were comprised of 100,000 lumpy background images and 400,000 CLB background images for the considered

BKS detection-localization tasks. Additionally, a “semi-online learning” method in which the measurement noise was generated on-the-fly was employed to mitigate the over-fitting problem [123]. Specifically, when training CNNs, the training data were simulated by adding measurement noise that was generated on-the-fly to the finite number of noiseless images [123]. In this way, the number of images employed to train the CNNs could be increased. Both the validation dataset and testing dataset comprised 200 images for each class.

Specifications of CNN architectures that possess the ability to approximate the posterior probability $\Pr(H_j|\mathbf{g})$ are required. A family of CNNs that comprise different number of convolutional (CONV) layers was explored to specify the CNN architecture. Specifically, a CNN having an initial architecture was firstly trained by minimizing the average of the cross-entropy over the training dataset defined Eq. (4.4). CNNs having more CONV layers were subsequently trained until the average of the cross-entropy over the validation dataset did not have significant decrement. A cross-entropy decrement of at least 1% of that produced by the previous CNN architecture was considered significant. The CNN that produced the minimum cross-entropy evaluated on the validation dataset was selected. All CNN architectures in the considered architecture family comprised CONV layers having 32 filters with the dimension of 5×5, a max-pooling layer [93], and a fully connected layer. A LeakyReLU activation function [99] was applied to the feature maps produced by each CONV layer and a softmax function was applied to the output of the fully connected layer. An instance of the considered CNN architecture is illustrated in Fig. 4.4. This architecture family was determined heuristically and may not be optimal for many other tasks. At each iteration of the training, the CNN weight parameters were updated by minimizing the empirical error function on mini-batches by use of the Adam algorithm [63], which is a stochastic gradient-based method.

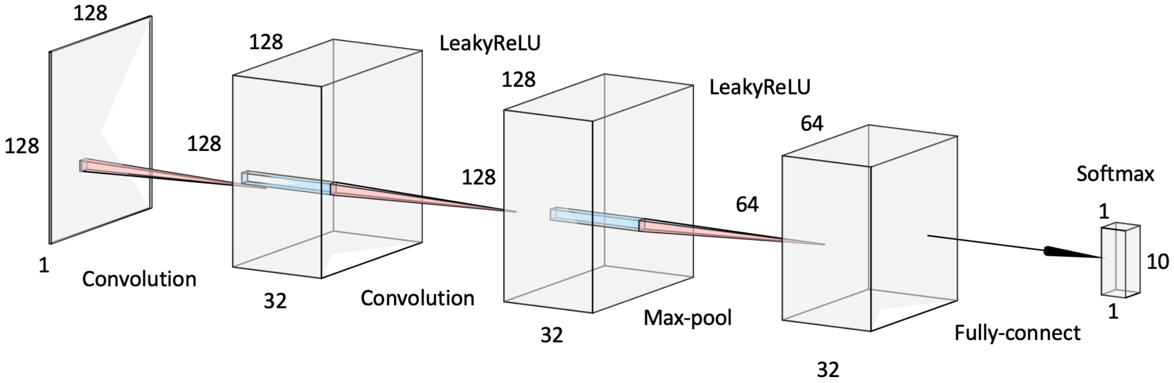


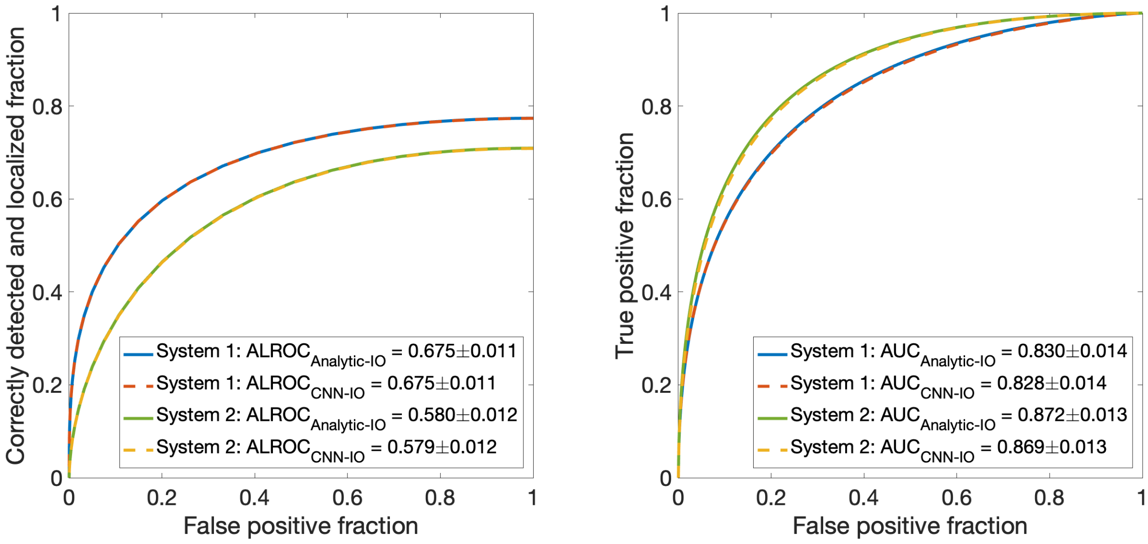
Figure 4.4: An instance of the CNN architecture for approximating a set of posterior probabilities for maximizing the ALROC. (© IEEE 2020)

4.4 Results

4.4.1 BKE signal detection-localization task

Convolutional neural networks that comprised one, three, and five CONV layers were trained for 500,000 mini-batches with each mini-batch comprising 80 images for each class. For both “System 1” and “System 2”, the validation cross-entropy was not significant decreased after 5 CONV layers were employed in the CNNs. Accordingly, we stopped training CNNs with more CONV layers, and the CNN corresponding to the smallest validation cross-entropy was selected, which was the CNN having five CONV layer.

For the joint detection-localization task, with both imaging systems, the LROC curves produced by the analytical computation (solid curves) are compared to those produced by the CNN (dashed curves) in Fig. 4.5 (a). In addition, for the simplified binary signal detection tasks, the ROC curves produced by the analytical computation (solid curves) are compared to those produced by the CNN (dashed curves) in Fig. 4.5 (b). The curves corresponding to the analytically computed IO and the CNN approximation of the IO (CNN-IO) are in close agreement in both cases. As shown in Fig. 4.5, the rankings of the two imaging systems are different when the joint detection-localization task and the simplified binary signal detection task were considered. When the signal detection-localization task is considered, “system 1” > “system 2”, while if the binary signal detection task is considered, “system 2” > “system 1”.



(a) (b)

Figure 4.5: (a) LROC curves corresponding to the IO for the BKE signal detection-localization tasks. (b) ROC curves corresponding to the IO for the simplified binary signal detection tasks. (© IEEE 2020)

4.4.2 BKS signal detection-localization task with a lumpy background model

Convolutional neural networks comprising 1, 3, 5, 7, 9, and 11 CONV layers were trained for 500,000 mini-batches with each mini-batch comprising 80 images for each class. The validation cross-entropy value was not significantly decreased after 11 CONV layers were employed in the CNN, and therefore the CNN having 11 CONV layers was selected for approximating the IO. The performance of the CNN for the signal detection-localization task was characterized by the LROC curve that was evaluated on the testing dataset. Note that the ALROC value produced by the CNN-IO was 0.711 ± 0.011, which was larger than the 0.530 ± 0.012 produced by the scanning HO.

The MCMC simulation provided further validation of the CNN-IO. The LROC curve produced by the MCMC method (blue curve) is compared to that produced by the CNN-IO (red-dashed curve) in Fig. 4.6. The curves are in close agreement. The ALROC values were 0.713 ± 0.011 and 0.711 ± 0.011 corresponding to the MCMC and the CNN-IO, respectively.

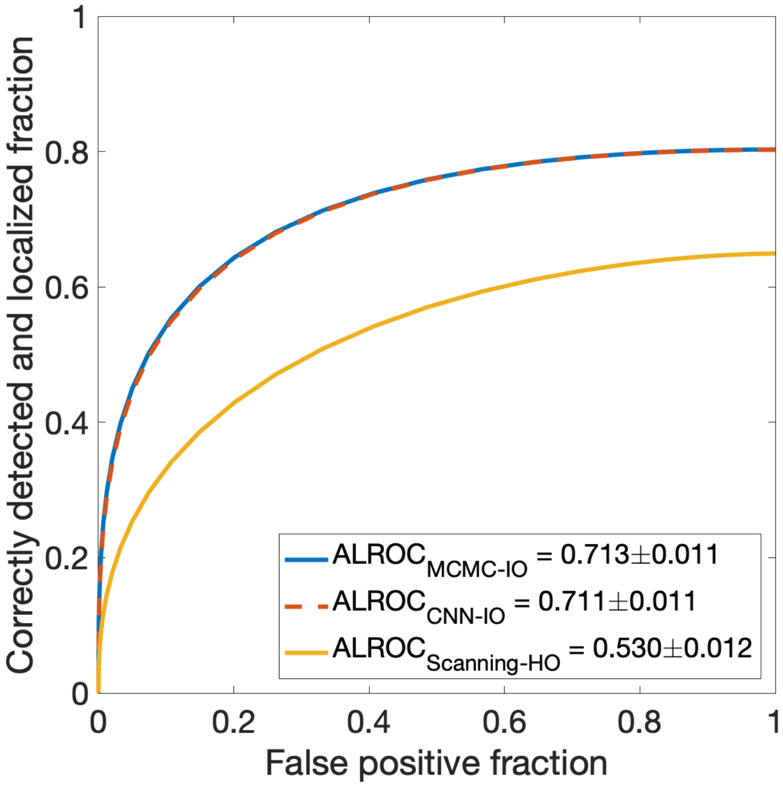


Figure 4.6: The LROC curves produced by the MCMC-IO (blue), CNN-IO (red-dashed), and the scanning HO (yellow) for the BKS task with the lumpy background model. The LROC curve corresponding to the CNN-IO closely approximates that corresponding to the MCMC-IO and is higher than that produced by the scanning HO. (© IEEE 2020)

4.4.3 BKS signal detection-localization task with a CLB model

Convolutional neural networks that comprised 1, 3, 5, and 7 CONV layers were trained for 500,000 mini-batches with each mini-batch comprising 20 images for each class. The validation cross-entropy value was not significantly decreased after 7 CONV layers were employed in the CNN, and therefore the CNN having 7 CONV layers was selected for approximating the IO. The performance of the selected CNN was quantified by computing the LROC curve and ALROC value on the testing dataset. The CNN-IO was compared to the scanning HO. The ALROC value produced by the CNN-IO was 0.749 ± 0.010, which was larger than the 0.637 ± 0.012 produced by the scanning HO as expected. The LROC curves corresponding to the CNN-IO and the scanning HO are displayed in Fig. 4.7.

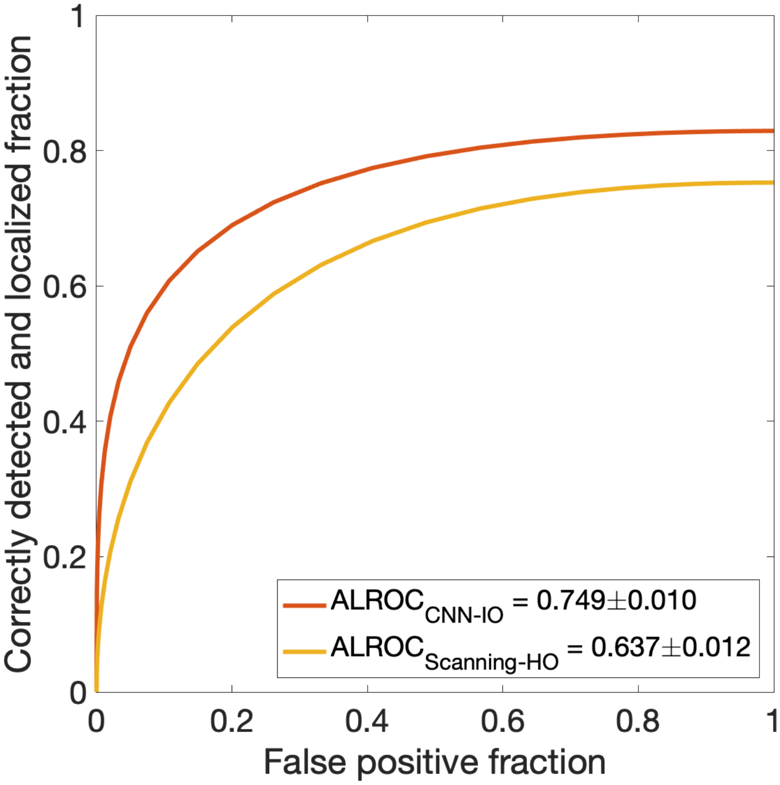


Figure 4.7: The LROC curves produced by CNN-IO (red) and the scanning HO (yellow) for the BKS task with the CLB model. As expected, the LROC curve corresponding to the CNN-IO is higher than that produced by the scanning HO. (© IEEE 2020)

Because the computation of the IO test statistic has not been addressed by MCMC methods for CLB models, validation corresponding to MCMC methods was not provided in this case.

4.5 Summary

Signal detection-localization tasks are of high interest when optimizing medical imaging systems and scanning numerical observers have been proposed to address them. However, there remains a scarcity of methods that can be implemented readily for approximating the IO for detection-localization tasks. In this chapter, a deep-learning-based method was investigated to address this need. Specifically, the proposed method provides a generalized framework for approximating the IO test statistic for multi-class classifications tasks. Compared to methods that employ MCMC techniques, supervised learning methods may be easier to implement. To properly run MCMC methods, numerous practical issues such as the design of proposal densities from which the Markov chain can be efficiently generated need to be addressed. Because of this, current applications of MCMC methods have been limited to relative simple object models such as a lumpy object model and a binary texture model. As such, the proposed supervised learning methods may possess a larger domain of applicability for approximating the IO than the MCMC methods. To demonstrate this, the proposed supervised learning method was applied to approximate the IO for a clustered lumpy object model, for which the IO approximation has not been achieved by the current MCMC methods.

The proposed supervised learning-based method may require a large amount of training data to accurately approximate the IO. Such data may be available when optimizing imaging systems and data acquisition designs via computer-simulation. In order to conduct a realistic computer-simulation, it is desirable to simulate images that capture anatomical variations and textures within a realistic object ensemble. To achieve this, one may establish a stochastic object model (SOM) from experimental data by training an AmbientGAN [15, 120, 121, 122]. Having a well-established SOM, one can produce large amount of training samples to train CNNs to accurately approximate the IO by use of the proposed supervised learning method. Such computer-simulation studies enable the exploration and assessment of a variety of imaging systems and data acquisition designs.

There remains several other topics for future investigation. It will be important to quantify the effect of the number of data used in the proposed method on the IO approximation. In addition, to implement the proposed supervised learning methods for approximating the IO in situations where only a limited number of experimental data is available, it will be important to investigate methods to train deep neural networks on limited training data. To achieve this, one may investigate the methods that employ domain adaptation [39, 49] and transfer learning [88]. Finally, it will be important to investigate supervised learning methods for approximating IOs for other more general tasks such as joint signal detection and estimation tasks associated with the estimation ROC (EROC) curve.

Chapter 5

Learning stochastic object models (SOMs) from medical imaging measurements using Progressively-Growing AmbientGANs

5.1 Introduction

Computer-simulation remains an important approach for the design and optimization of imaging systems. Such approaches can permit the exploration, refinement, and assessment of a variety of system designs that would be infeasible through experimental studies alone. In the field of medical imaging, it has been advocated that imaging systems and reconstruction algorithms should be assessed and optimized by use of objective measures of image quality (IQ) that quantify the performance of an observer at specific diagnostic tasks [5, 9, 11, 80, 104]. To accomplish this, all sources of variability in the measured data should be accounted for. One important source of variability that can significantly limit observer performance is variation in the objects to-be-imaged [91]. This source of variability can be described by stochastic object models (SOMs) [66]. A SOM is a generative model that can be employed to produce an ensemble of to-be-imaged objects that possess prescribed statistical properties.

Available SOMs include texture models of mammographic images with clustered lumpy backgrounds [13], simple lumpy background models [91], and more realistic anatomical phantoms that can be randomly perturbed [95]. A variety of other computational phantoms [19, 28, 72, 94, 95, 109, 112, 126], either voxelized or mathematical, have been proposed for medical imaging simulation, aiming to provide a practical solution to characterize object variability. However, the majority of these were established by use of image data corresponding to a few subjects. Therefore, they may not accurately describe the statistical properties of the ensemble of objects that is relevant to an imaging system optimization task. A variety of anatomical shape models have also been proposed to describe both the common geometric features and the geometric variability among instances of the population for shape analysis applications [4, 29, 30, 37, 47, 51, 97, 103]. To date, these have not been systematically explored for the purpose of constructing SOMs that capture realistic anatomical variations for use in imaging system optimization.

In order to establish SOMs that capture realistic textures and anatomical variations, it is desirable to utilize experimental imaging data. By definition, however, SOMs should be independent of the imaging system, measurement noise and any reconstruction method employed. In other words, they should provide an *in silico* representation of the ensemble of objects to-be-imaged and not estimates of them that would be indirectly measured or computed by imaging systems. To address this need, Kupinski *et al.* [66] proposed an explicit generative model for describing object statistics that was trained by use of noisy imaging measurements and a computational model of a well-characterized imaging system [66]. However, applications of this method have been limited to situations where the characteristic function of the corresponding imaging measurements can be analytically determined, such as with lumpy and clustered lumpy object models [13, 64]. As such, there remains a need to generalize the method so that anatomically realistic and more complicated SOMs can be established from experimental imaging measurements.

Deep generative neural networks, such as generative adversarial networks (GANs) [46], hold great potential for establishing SOMs that describe discretized objects. However, conventional GANs are typically trained by use of reconstructed images that are influenced by the effects of measurement noise and the reconstruction process. To circumvent this, an AmbientGAN has been proposed [15] that augments a GAN with a measurement operator. This permits a generative model that describes object randomness to be learned from indirect and noisy measurements of the objects themselves. In a preliminary study, the AmbientGAN was explored for the establishing SOMs from imaging measurements for use in optimizing imaging systems [122]. However, similar to conventional GANs, the process of training AmbientGANs is inherently unstable. Moreover, the original AmbientGAN cannot immediately benefit from robust GAN training procedures, such as progressive growing [59], which limits its ability to synthesize high-dimensional images that depict objects of interest in medical imaging studies.

In this chapter, a new AmbientGAN approach is proposed that permits the utilization of the progressive growing strategy for training. In this way, SOMs can be established from noisy imaging measurements that can yield high-dimensional images that depict objects. The new approach, referred to as a Progressive Growing AmbientGAN (ProAmGAN), can utilize the progressive growing training strategy due to augmentation of the conventional AmbientGAN architecture with an image reconstruction operator. Stylized numerical studies corresponding to X-ray computed tomography (CT) and magnetic resonance (MR) imaging are conducted to investigate the proposed ProAmGAN for establishing SOMs. Preliminary validation studies are presented that utilize standard quantitative measures for evaluating GANs and also objective measures based on signal detection performance.

5.2 Background

Consider a discrete-to-discrete (D-D) description of a linear imaging system given by [9]:

$$\mathbf{g} = \mathbf{H}\mathbf{f} + \mathbf{n}, \tag{5.1}$$

where $\mathbf{g} \in \mathbb{R}^M$ is a vector that describes the measured image data, $\mathbf{f} \in \mathbb{R}^N$ denotes the finite-dimensional representation of the object being imaged, $\mathbf{H} \in \mathbb{R}^{M\times N}$ denotes a D-D imaging operator $\mathbb{R}^N \to \mathbb{R}^M$ that maps an object in the Hilbert space $\mathbb{U}$ to the measured discrete data in the Hilbert space $\mathbb{V}$, and the random vector $\mathbf{n} \in \mathbb{R}^M$ denotes the measurement noise. Below, the imaging process described in Eq. (5.1) is denoted as: $\mathbf{g} = \mathcal{H}_{\mathbf{n}}(\mathbf{f})$. It is assumed that the D-D imaging model is a sufficiently accurate representation of the true continuous-to-discrete (C-D) imaging model that describes a digital imaging system and the impact of model error will be neglected. When optimizing imaging system performance by use of objective measures of IQ, all sources of randomness in $\mathbf{g}$ should be considered. In diagnostic imaging applications, object variability is an important factor that limits observer performance. In such applications, the object $\mathbf{f}$ should be described as a random vector that is characterized by a multivariate probability density function (PDF) $\mathrm{pr}(\mathbf{f})$ that specifies the statistical properties of the ensemble of objects to-be-imaged.

Direct estimation of $\mathrm{pr}(\mathbf{f})$ is rarely tractable in medical imaging applications due to the high dimensionality of $\mathbf{f}$. To circumvent this difficulty, a parameterized generative model, referred to throughout this work as a SOM, can be introduced and established by use of an ensemble of experimental measurements. The generative model can be explicit or implicit. Explicit generative models seek to approximate pr($\mathbf{f}$), or equivalently, its characteristic function, from which samples $\mathbf{f}$ can subsequently be drawn. On the other hand, implicit generative models do not seek to estimate pr($\mathbf{f}$) directly, but rather define a stochastic process that seeks to draw samples from pr($\mathbf{f}$) without having to explicitly specify it. Variational autoencoders and GANs are examples of explicit and implicit generative models, respectively, that have been actively explored [45]. Two previous works that sought to learn SOMs from noisy and indirect imaging measurements by use of explicit and implicit generative models are presented below.

5.2.1 Establishing SOMs by use of explicit generative modeling: Propagation of characteristic functionals

The first method to learn SOMs from imaging measurements was introduced by Kupinski *et al.* [66]. In that work, a C-D imaging model was considered in which a function that describes the object is mapped to a finite-dimensional image vector $\mathbf{g}$. For C-D operators, it has been demonstrated that the characteristic functional (CFl) describing the object can be readily related to the characteristic function (CF) of the measured data vector $\mathbf{g}$ [26]. This provides a relationship between the PDFs of the object and measured image data. In their method, an object that was parameterized by the vector $\boldsymbol{\Theta}$ was considered and analytic expressions for the CFl were utilized. Subsequently, by use of the known imaging operator and noise model, the corresponding CF was computed. The vector $\boldsymbol{\Theta}$ was estimated by minimizing the discrepancy between this model-based CF and an empirical estimate of the CF computed from an ensemble of noisy imaging measurements. From the estimated CFl, an ensemble of objects could be generated. This method was applied to establish SOMs where the CFl of the object can be analytically determined. Such cases include the lumpy object model [64] and clustered lumpy object model [13]. The applicability of the method to more complicated object models remains unexplored.

5.2.2 Establishing SOMs by use of implicit generative modeling: Generative adversarial networks (GANs) and AmbientGANs

Generative adversarial networks (GANs) [6, 7, 8, 17, 32, 46, 48, 73, 89, 92, 98] are implicit generative models that have been actively explored to learn the statistical properties of ensembles of images and generate new images that are consistent with them. A traditional GAN consists of two deep neural networks—a generator and a discriminator. The generator is jointly trained with the discriminator through an adversarial process. During its training process, the generator is trained to map random low-dimensional latent vectors to higher dimensional images that represent samples from the distribution of training images. The discriminator is trained to distinguish the generated, or synthesized, images from the actual training images. These are often referred to as the “fake” and “real” images in the GAN literature. Subsequent to training, the discriminator is discarded and the generator and associated latent vector probability distribution form as an implicit generative model that can sample from the data distribution to produce new images. However, images produced by imaging systems are contaminated by measurement noise and potentially an image reconstruction process. Therefore, GANs trained directly on images do not generally represent SOMs because they do not characterize object variability alone.

An augmented GAN architecture named AmbientGAN has been proposed [15] that enables learning an SOM from noisy indirect measurements of an object. As shown in Fig. 5.1, the AmbientGAN architecture includes the measurement operator $\mathcal{H}_{\mathbf{n}}$, defined in Eq. (5.1), into the traditional GAN framework. During the AmbientGAN training process, the generator is

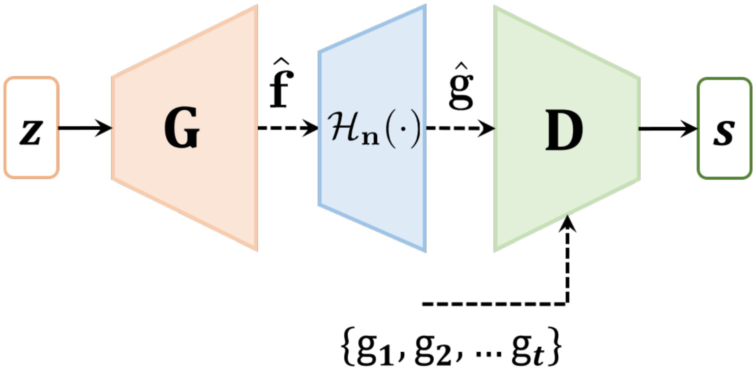


Figure 5.1: An illustration of the AmbientGAN architecture. The generator **G** is trained to generate objects, which are subsequently employed to simulate measurement data. The discriminator **D** is trained to distinguish "real" measurement data to the "fake" measurement data that are simulated by use of the generated objects.

trained to map a random vector $\mathbf{z} \in \mathbb{R}^k$ described by a latent probability distribution to a generated object $\hat{\mathbf{f}} = G(\mathbf{z}; \mathbf{\Theta}_G)$, where $G : \mathbb{R}^k \to \mathbb{R}^N$ represents the generator network that is parameterized by a vector of trainable parameters $\mathbf{\Theta}_G$. Subsequently, the corresponding simulated imaging measurements are computed as $\hat{\mathbf{g}} = \mathcal{H}_{\mathbf{n}}(\hat{\mathbf{f}})$. The discriminator neural network $D : \mathbb{R}^N \to \mathbb{R}$, which is parameterized by the vector $\mathbf{\Theta}_D$, is trained to distinguish the real and simulated imaging measurements by mapping them to real-valued scalar s. The adversarial training process can be represented by the following two-player minimax game [46]:

$$\min_{\mathbf{\Theta}_G} \max_{\mathbf{\Theta}_D} V(D, G) = E_{\mathbf{g} \sim p_{\mathbf{g}}}[l\left(D(\mathbf{g}; \mathbf{\Theta}_D)\right)] + E_{\hat{\mathbf{g}} \sim p_{\hat{\mathbf{g}}}}[l(1 - D\left(\hat{\mathbf{g}}; \mathbf{\Theta}_D\right))], \tag{5.2}$$

where $l(\cdot)$ represents a loss function. When the distribution of objects $\text{pr}(\mathbf{f})$ uniquely induces the distribution of imaging measurements $\text{pr}(\mathbf{g})$, i.e., when the imaging operator is injective, and the minimax game achieves the global optimum, the trained generator can be employed to produce object samples drawn from $\text{pr}(\mathbf{f})$ [15, 46].

Zhou *et al.* have demonstrated the ability of the AmbientGAN to learn a simple SOM corresponding to a lumpy object model that could be employed to produce small (64×64)

object samples [122]. However, adversarial training is known to be unstable and the use of AmbientGANs to establish realistic and large-scale SOMs has, to-date, been limited.

5.2.3 Progressively-Growing GAN Training Strategy

A novel training strategy for GANs—progressive growing of GANs (ProGANs)—has been recently developed to improve the stability of the GAN training process [59] and hence the ability to learn generators that sample from distributions of high-resolution images. GANs are conventionally trained directly on full size images through the entire training process. In contrast, ProGANs adopt a multi-resolution approach to training. Initially, a generator and discriminator are trained by use of down-sampled (low resolution) training images. During each subsequent training stage, higher resolution versions of the original training images are employed to train progressively deeper discriminators and generators, continuing until a final version of the generator is trained by use of the original high-resolution images. While this progressively growing training strategy has found widespread success with conventional GANs, as described below, it cannot generally be employed with AmbientGANs. A solution to this problem is described next.

5.3 Establishing SOMs by use of Progressively-Growing AmbientGANs

As discussed above, AmbientGANs enable the learning of SOMs from noisy imaging measurements but can be difficult to train, while ProGANs can be stably trained and established by use of higher-dimensional image data that are generally affected by noise and the image formation process. Below, a novel strategy, Progressively Growing AmbientGANs (ProAmGANs), is proposed to enable progressive growing of AmbientGANs for learning realistic SOMs from noisy and indirect imaging measurements.

The ProAmGAN progressively grows the generator to establish the SOM from its low-resolution version to full-resolution version. As with the AmbientGANs, the imaging measurements are subsequently simulated by applying the measurement operator to the generator-produced objects. However, imaging measurements acquired in most medical imaging systems are indirect representations of objects to-be-imaged (e.g., Radon transform data, k-space data). In such cases, the low-resolution version of the measured image data and the low-resolution version of the objects may not be simply related because they reside in generally different Hilbert spaces. Accordingly, in these cases, the progressive growing strategy cannot be directly applied because the generator in the original ProGAN produces images that reside in the same Hilbert space as the training data employed by the discriminator. To address this issue, in addition to including the measurement operator as with the AmbientGAN training strategy, an image reconstruction operator $\mathcal{O}$: $\mathbb{R}^M \to \mathbb{R}^N$ is included in the proposed ProAmGAN training strategy. In this way, the generator can be trained to produce images that reside in the same Hilbert space as the images employed by the discriminator and the progressive growing strategy can be subsequently employed. The ProAmGAN training strategy is illustrated in Fig. 5.2.

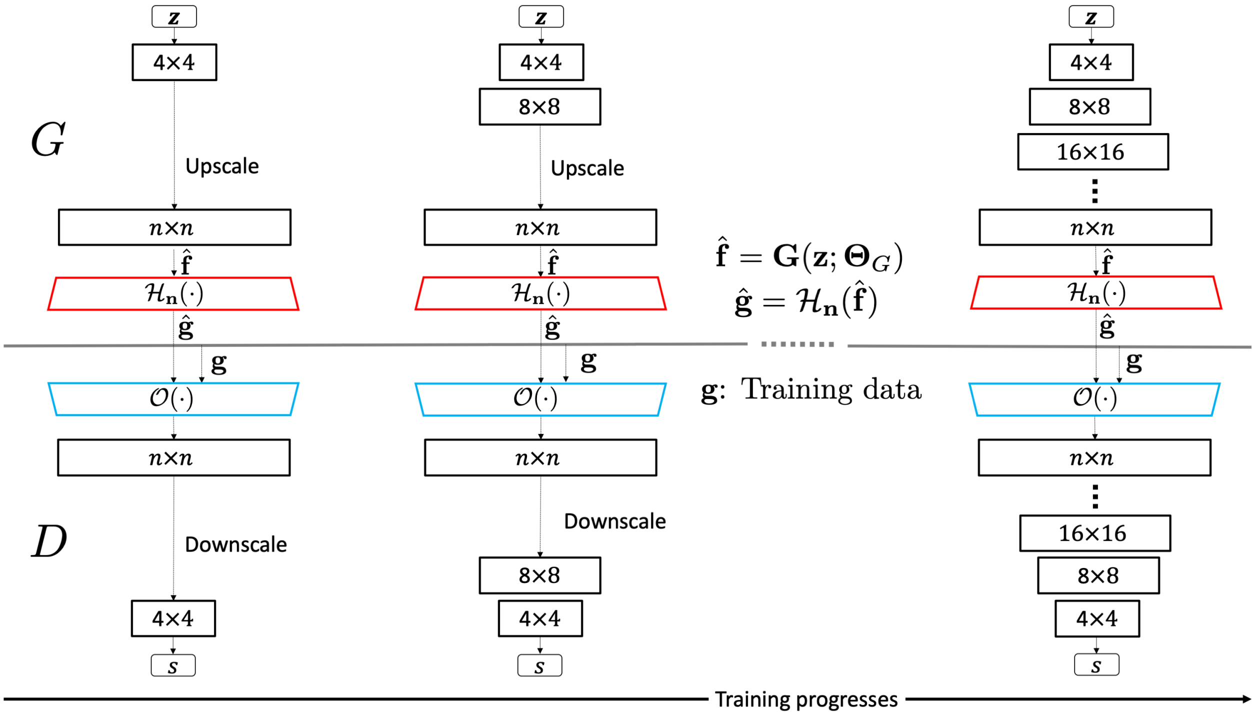


Figure 5.2: An illustration of ProAmGAN training. The training starts with low image resolution (e.g., 4×4) and the image resolution is increased progressively by adding more layers to the generator and the discriminator. The discriminator is trained to distinguish between the ground-truth and generated reconstructed objects.

Given a training dataset that comprises measured data $\mathbf{g}$, a set of reconstructed objects $\mathbf{f}_{recon}$ is computed by applying the operator $\mathcal{O}$ to the measured data $\mathbf{g}$: $\mathbf{f}_{recon} = \mathcal{O}(\mathbf{g}) \equiv \mathcal{O}(\mathcal{H}_{\mathbf{n}}(\mathbf{f}))$. Denote the reconstructed object corresponding to the generator-produced measured data $\hat{\mathbf{g}}$ as $\hat{\mathbf{f}}_{recon}$: $\hat{\mathbf{f}}_{recon} = \mathcal{O}(\hat{\mathbf{g}}) \equiv \mathcal{O}\Big(\mathcal{H}_{\mathbf{n}}\big(\mathbf{G}(\mathbf{z}; \mathbf{\Theta}_G)\big)\Big)$. The discriminator in the ProAmGAN is trained to distinguish between $\hat{\mathbf{f}}_{recon}$ and $\mathbf{f}_{recon}$, and the generator is trained to generate objects $\hat{\mathbf{f}} = \mathbf{G}(\mathbf{z}; \mathbf{\Theta}_G)$ such that the corresponding reconstructed objects $\hat{\mathbf{f}}_{recon}$ are indistinguishable from the reconstructed objects $\mathbf{f}_{recon}$ that were reconstructed from the provided measurement data (i.e., training data). As with the AmbientGAN, when the distribution of objects $\mathrm{pr}(\mathbf{f})$ uniquely induces the distribution of reconstructed objects $\mathrm{pr}(\mathbf{f}_{recon})$, and the ProAmGAN achieves the global optimal at the final full-resolution stage, the trained generator can be employed to produce object samples drawn from the distribution $\mathrm{pr}(\mathbf{f})$. In special cases where the imaging operator $\mathbf{H}$ is full-rank and the measurement noise $\mathbf{n} = \mathbf{0}$, ProAmGANs reduce to original ProGANs that are directly trained on objects.

5.4 Numerical studies

Computer-simulation studies were conducted to demonstrate the ability of the proposed ProAmGAN to establish realistic SOMs from imaging measurements corresponding to different stylized imaging modalities. Details regarding the design of the computer-simulation studies are provided below.

5.4.1 Idealized direct imaging system

An idealized direct imaging system that acquired chest radiographs, modeled as: $\mathbf{g} = \mathbf{f} + \mathbf{n}$, was considered first. By design, it was assumed that the measurement noise was the only source of image degradation. The motivation for this study was to demonstrate the ability of the ProAmGAN to learn an SOM from noisy images.

An NIH database of clinical chest X-ray images [105] was employed to serve as ground truth objects $\mathbf{f}$. Three thousand images were selected from this dataset. These images were centrally cropped and resized to the dimension of 512×512 and were normalized to the range between 0 and 1. A collection of 3000 simulated measured images $\mathbf{g}$ were produced by adding independent and identically distributed (i.i.d.) Gaussian noise with zero mean and the standard deviation of 2% to the collection of objects $\mathbf{f}$. An example of the objects and the corresponding noisy imaging measurement are shown in Fig. 5.3.

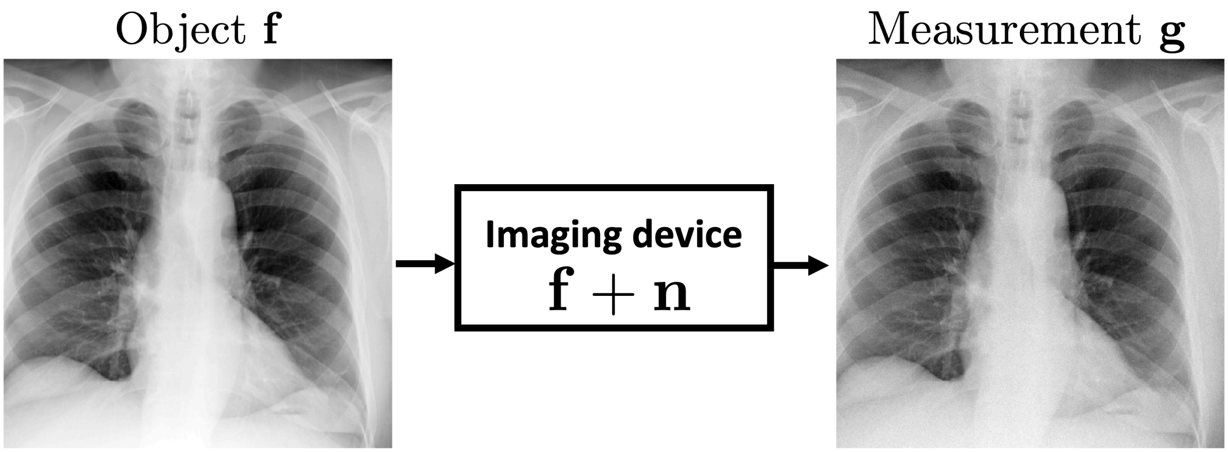


Figure 5.3: An illustration of idealized planar X-ray imaging system that acquires noisy imaging measurements.

From the ensemble of simulated measured data, with the knowledge of the measurement noise model, a ProAmGAN was trained to establish a SOM that characterizes the distribution of objects $\mathbf{f}$. The architecture of the generator and the discriminator employed in the ProAmGAN is described in Table 5.1 (a). Because the idealized planar X-ray imaging system acquires direct representations of objects (i.e., $\mathbb{V} = \mathbb{U}$), the reconstruction operator $\mathcal{O}(\cdot)$ was set to be the identity operator in the ProAmGAN training process.

For comparison, by use of the same ensemble of simulated measured images $\mathbf{g}$, a ProGAN was trained. In this case, the generator was trained to learn the distribution of measured images $\mathbf{g}$ themselves, which are contaminated by measurement noise, instead of learning the distribution of objects $\mathbf{f}$ (i.e., the SOM). The ProGAN employed a generator and discriminator with the same architectures as those employed in the ProAmGAN.

Generator	Act.	Output shape
Latent vector	-	512×1×1
Conv 4×4	LReLU	512×4×4
Conv 3×3	LReLU	512×4×4
Upscale	-	512×8×8
Conv 3×3	LReLU	512×8×8
Conv 3×3	LReLU	512×8×8
Upscale	-	512×16×16
Conv 3×3	LReLU	512×16×16
Conv 3×3	LReLU	512×16×16
Upscale	-	512×32×32
Conv 3×3	LReLU	512×32×32
Conv 3×3	LReLU	512×32×32
Upscale	-	512×64×64
Conv 3×3	LReLU	256×64×64
Conv 3×3	LReLU	256×64×64
Upscale	-	256×128×128
Conv 3×3	LReLU	128×128×128
Conv 3×3	LReLU	128×128×128
Upscale	-	128×256×256
Conv 3×3	LReLU	64×256×256
Conv 3×3	LReLU	64×256×256
Upscale	-	64×512×512
Conv 3×3	LReLU	32×512×512
Conv 3×3	LReLU	32×512×512
Conv 1×1	linear	1×512×512

Discriminator	Act.	Output shape
Input image	-	1 × 512 × 512
Conv 1 × 1	LReLU	32 × 512 × 512
Conv 3 × 3	LReLU	32 × 512 × 512
Conv 3 × 3	LReLU	64 × 512 × 512
Downscale	-	64 × 256 × 256
Conv 3 × 3	LReLU	64 × 256 × 256
Conv 3 × 3	LReLU	128 × 256 × 256
Downscale	-	128 × 128 × 128
Conv 3 × 3	LReLU	128 × 128 × 128
Conv 3 × 3	LReLU	256 × 128 × 128
Downscale	-	256 × 64 × 64
Conv 3 × 3	LReLU	256 × 64 × 64
Conv 3 × 3	LReLU	512 × 64 × 64
Downscale	-	512 × 32 × 32
Conv 3 × 3	LReLU	512 × 32 × 32
Conv 3 × 3	LReLU	512 × 32 × 32
Downscale	-	512 × 16 × 16
Conv 3 × 3	LReLU	512 × 16 × 16
Conv 3 × 3	LReLU	512 × 16 × 16
Downscale	-	512 × 8 × 8
Conv 3 × 3	LReLU	512 × 8 × 8
Conv 3 × 3	LReLU	512 × 8 × 8
Downscale	-	512 × 4 × 4
Minibatch stddev	-	513 × 4 × 4
Conv 3 × 3	LReLU	512 × 4 × 4
Conv 4 × 4	LReLU	512 × 1 × 1
Fully-connected	linear	1 × 1 × 1

(a)

Generator	Act.	Output shape
Latent vector	-	512×1×1
Conv 4×4	LReLU	512×4×4
Conv 3×3	LReLU	512×4×4
Upscale	-	512×8×8
Conv 3×3	LReLU	512×8×8
Conv 3×3	LReLU	512×8×8
Upscale	-	512×16×16
Conv 3×3	LReLU	512×16×16
Conv 3×3	LReLU	512×16×16
Upscale	-	512×32×32
Conv 3×3	LReLU	512×32×32
Conv 3×3	LReLU	512×32×32
Upscale	-	512×64×64
Conv 3×3	LReLU	256×64×64
Conv 3×3	LReLU	256×64×64
Upscale	-	256×128×128
Conv 3×3	LReLU	128×128×128
Conv 3×3	LReLU	128×128×128
Upscale	-	128×256×256
Conv 3×3	LReLU	64×256×256
Conv 3×3	LReLU	64×256×256
Conv 1×1	linear	1×256×256

Discriminator	Act.	Output shape
Input image	-	1 × 256 × 256
Conv 1 × 1	LReLU	64 × 256 × 256
Conv 3 × 3	LReLU	64 × 256 × 256
Conv 3 × 3	LReLU	128 × 256 × 256
Downscale	-	128 × 128 × 128
Conv 3 × 3	LReLU	128 × 128 × 128
Conv 3 × 3	LReLU	256 × 128 × 128
Downscale	-	256 × 64 × 64
Conv 3 × 3	LReLU	256 × 64 × 64
Conv 3 × 3	LReLU	512 × 64 × 64
Downscale	-	512 × 32 × 32
Conv 3 × 3	LReLU	512 × 32 × 32
Conv 3 × 3	LReLU	512 × 32 × 32
Downscale	-	512 × 16 × 16
Conv 3 × 3	LReLU	512 × 16 × 16
Conv 3 × 3	LReLU	512 × 16 × 16
Downscale	-	512 × 8 × 8
Conv 3 × 3	LReLU	512 × 8 × 8
Conv 3 × 3	LReLU	512 × 8 × 8
Downscale	-	512 × 4 × 4
Minibatch stddev	-	513 × 4 × 4
Conv 3 × 3	LReLU	512 × 4 × 4
Conv 4 × 4	LReLU	512 × 1 × 1
Fully-connected	linear	1 × 1 × 1

(b)

Table 5.1: The architectures of the generator and discriminator for generating 512 × 512 images (a) and those for generating 256 × 256 images (b). More details about each component in the architecture can be found in ProGAN paper [59].

The Fréchet Inception Distance (FID) [53] score, a widely employed metric to evaluate the performance of generative models, was computed to evaluate the performance of the original ProGAN and the proposed ProAmGAN. The FID score quantifies the distance between the features extracted by the Inception-v3 network [102] from the ground-truth ("real") and generated objects (“fake”). Lower FID score indicates better quality and diversity of the generated objects. The FID scores were computed by use of 3000 ground-truth objects, 3000 ProGAN-generated objects and 3000 ProAmGAN-generated objects.

The structural similarity index (SSIM) [106] is a figure-of-merit describing the similarity of two digital images. As another form of evaluation, SSIM values were computed for different pairs of images. First, SSIM values were computed from 500,000 random pairs of ground truth objects. Next, SSIM values were computed from 500,000 random pairs of ProAmGAN-generated and ground truth objects. Finally, as a comparison, SSIM values were computed from 500,000 random pairs of ProGAN-generated and ground truth objects. From these three collections of SSIM values, three histograms were formed. The overlap area between any two of the histograms (i.e., empirical PDFs) and the two-sample Kolmogorov-Smirnov (KS) test statistics [111] were computed.

5.4.2 Stylized computed tomographic imaging system

A stylized tomographic imaging system was investigated next. This imaging system was described as: $\mathbf{g} = \mathcal{R}\mathbf{f} + \mathbf{n}$, where $\mathcal{R}$ denotes a 2D discrete Radon transform [58] that maps a 2D object $\mathbf{f}$ to a sinogram. The angular scanning range was 180 degrees and tomographic views were evenly spaced with a 1 degree angular step.

An NIH-sponsored database of clinical chest CT images [110] was employed to serve as ground truth objects $\mathbf{f}$. Three thousand images of dimension of 512×512 were selected from this dataset and were normalized to the range between 0 and 1. A collection of 3000 measured data $\mathbf{g}$ were simulated by acting $\mathcal{R}$ on each object and adding i.i.d. Gaussian noise with a standard deviation of 10%. An example of the objects and the corresponding measured imaging data are shown in Fig. 5.4.

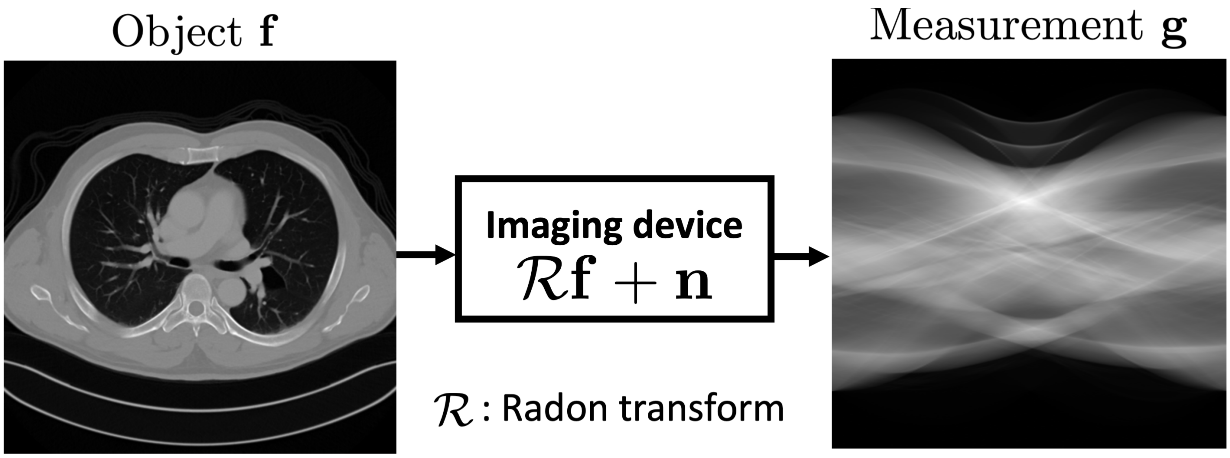


Figure 5.4: An illustration of tomographic imaging system that acquires Radon transform data.

From the collection of measured data $\mathbf{g}$, a set of reconstructed objects $\mathbf{f}_{recon}$ was generated by use of a filtered back-projection (FBP) reconstruction algorithm that employed a Ram-Lak filter. With the knowledge of the imaging operator and the measurement noise model, a ProAmGAN was subsequently trained by use of the reconstructed objects. The ProAmGAN employed the generator and discriminator with the architectures described in Table 5.1 (a). In the ProAmGAN training process, the Radon transform $\mathcal{R}$ and the FBP operator were applied to the generated objects as discussed in Sec. 5.3.

As a comparison, a ProGAN was trained by use of reconstructed objects $\mathbf{f}_{recon}$. The generator in the ProGAN was trained to learn the distribution of $\mathbf{f}_{recon}$ instead of learning the distribution of $\mathbf{f}$. The ProGAN employed a generator and discriminator with the same architectures as those employed in the ProAmGAN. The FID scores and empirical PDFs of SSIM values were computed as described in Sec. 5.4.1.

5.4.3 Stylized magnetic resonance (MR) imaging system with complete k-space data

A stylized MR imaging system that acquires fully-sampled k-space data was investigated. This imaging system was described as: $\mathbf{g} = \mathcal{F}(\mathbf{f}) + \mathbf{n}$, where $\mathcal{F}$ denotes a 2D discrete Fourier transform (DFT). A database of clinical brain MR images [21] were employed to serve as ground truth objects $\mathbf{f}$. Three thousand images having the dimension of 512×512 were selected from this dataset and were normalized to the range between 0 and 1. A collection of 3000 measured image data $\mathbf{g}$ were simulated by computing the 2D DFT of the objects and adding i.i.d. zero mean Gaussian noise with a standard deviation of 10 to both the real and imaginary components. An example of the objects and the corresponding magnitude of the measured k-space data are shown in Fig. 5.5.

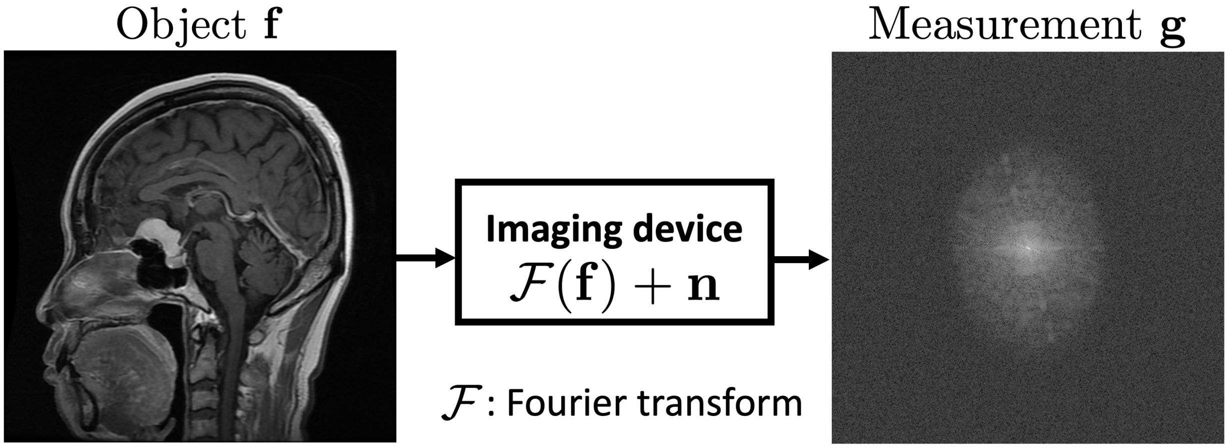


Figure 5.5: MR imaging system with complete k-space data. Logarithm of one plus the magnitude of k-space data was displayed.

From the ensemble of measured images, an ensemble of reconstructed images $\mathbf{f}_{recon}$ was generated by acting a 2D inverse discrete Fourier transform (IDFT) to each measured image data $\mathbf{g}$. A ProAmGAN was subsequently trained to establish a SOM that characterizes the distribution of objects $\mathbf{f}$ by use of the ensemble of reconstructed images $\mathbf{f}_{recon}$. The ProAmGAN employed a generator and discriminator with architectures described in Table 5.1 (a). In the training process, the 2D DFT and IDFT were applied to the generator-produced objects as discussed in Sec. 5.3.

For comparison, a ProGAN was trained by use of reconstructed images $\mathbf{f}_{recon}$. The ProGAN employed a generator and discriminator with the same architectures as those employed in the ProAmGAN. The FID score and empirical PDFs of SSIM values were also computed as described in Sec. 5.4.1.

5.4.4 Stylized MR imaging system with under-sampled k-space data

MR imaging systems sometimes acquire under-sampled k-space data to accelerate the data-acquisition process. In such cases, the imaging operator $\mathbf{H}$ has a non-trivial null space and only the measurement component $\mathbf{f}_{meas} = \mathbf{H}^{\dagger}\mathbf{H}\mathbf{f}$ can be observed through the imaging system. Here, $\mathbf{H}^{\dagger}$ denotes the Moore-Penrose pseudo-inverse of $\mathbf{H}$ and can be computed by applying a 2D IDFT to the zero-filled k-space data. In this study, the impact of k-space under-sampling on images produced by the ProAmGAN was investigated.

Clinical brain MR images contained in the NYU fastMRI Initiative database [113] were employed to serve as ground truth objects $\mathbf{f}$. Three thousand images having dimension of 320×320 were selected from this database for use in this study. These images were resized to the dimension of 256×256 and were normalized to the range between 0 and 1. Five data-acquisition designs corresponding to different k-space sampling ratios were considered: 1/1, 4/5, 1/2, 1/4, and 1/8. Here, the k-space sampling ratio was defined as the ratio of the number of sampled k-space components to the number of complete k-space components. The sampling patterns are illustrated in the top row of Fig. 5.6. For each considered design, a collection of 3000 measured data $\mathbf{g}$ were simulated by computing and sampling the k-space data and adding i.i.d. zero mean Gaussian noise with a standard deviation of 2 to both the real and imaginary components.

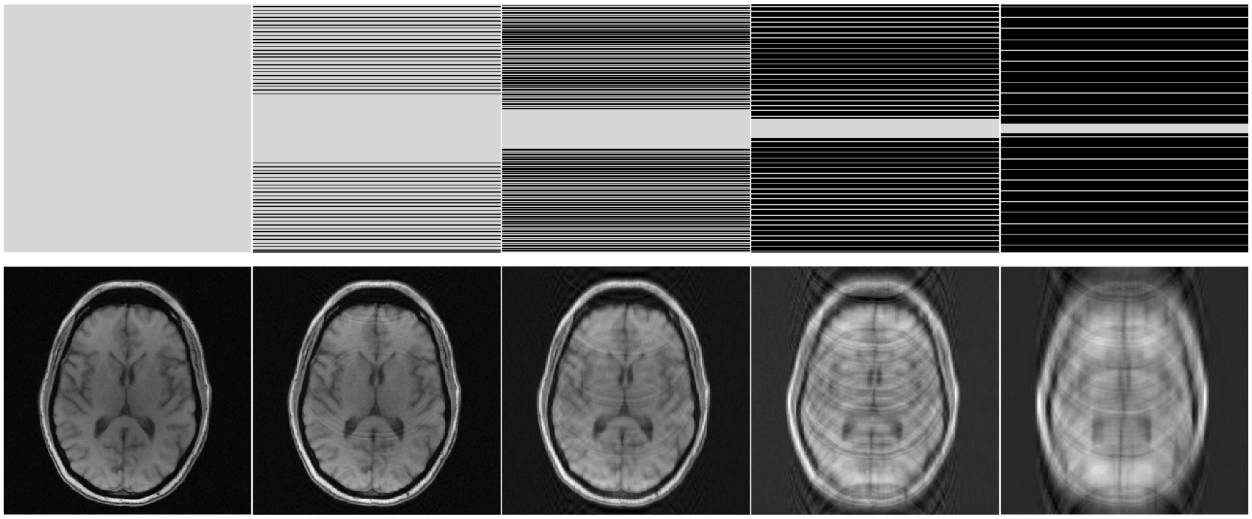


Figure 5.6: Top: k-space sampling patterns corresponding to different sampling ratios of 1/1, 4/5, 1/2, 1/4, and 1/8 from left to right; Bottom: images reconstructed by use of $\mathbf{H}^\dagger$ corresponding to the k-space sampling patterns in the top row.

For each data-acquisition design, reconstructed objects $\mathbf{f}_{recon}$ were produced by acting the pseudo-inverse operator $\mathbf{H}^\dagger$ on the given measured image data $\mathbf{g}$. Examples of reconstructed images using pseudo-inverse method corresponding to the considered sampling patterns are shown in the bottom row of Fig. 5.6. A ProAmGAN was subsequently trained to establish a SOM for each data-acquisition design. The architecture of the generator and the discriminator employed in the ProAmGAN is described in Table 5.1 (b). In the training process, $\mathbf{H}$ and $\mathbf{H}^\dagger$ were applied to the generator-produced objects as discussed in Sec. 5.3. The FID score was computed by use of 3000 ground-truth objects $\mathbf{f}$ and 3000 ProAmGAN-generated objects

$\hat{\mathbf{f}}$ for each data-acquisition design. Because only the measurement component $\mathbf{f}_{meas} = \mathbf{H}^{\dagger}\mathbf{H}\mathbf{f}$ can be measured by imaging systems, the ability of ProAmGANs to learn the variation in the measurement components was investigated. Specifically, the FID score was computed by use of the ground-truth measurement components $\mathbf{f}_{meas} = \mathbf{H}^{\dagger}\mathbf{H}\mathbf{f}$ and ProAmGAN-generated measurement components $\hat{\mathbf{f}}_{meas} = \mathbf{H}^{\dagger}\mathbf{H}\hat{\mathbf{f}}$ for each data-acquisition design.

As a comparison, an original ProGAN was trained by use of the reconstructed objects $\mathbf{f}_{recon}$ for each data-acquisition design. The ProGAN employed the generator and the discriminator with the same architecture as those employed in the ProAmGAN. The ProGAN-produced images were compared to the ProAmGAN-produced images.

5.4.5 Task-based image quality assessment

In this study, the ProAmGAN-established SOMs corresponding to fastMRI brain objects were evaluated by use of objective measures of IQ. Specifically, the ProAmGAN-established SOMs were evaluated by comparing task-specific image quality measures computed by use of generated objects to those computed by use of ground-truth objects. A signal-known-exactly binary classification task was considered in which an observer classifies noisy MR images as satisfying either a signal-absent hypothesis (H_0) or signal-present hypothesis (H_1). The imaging processes under these two hypotheses can be described as:

$$H_0 : \mathbf{g} = \mathbf{f} + \mathbf{n}, \tag{5.3a}$$

$$H_1 : \mathbf{g} = \mathbf{f} + \mathbf{s} + \mathbf{n}, \tag{5.3b}$$

where $\mathbf{s}$ denotes a signal image and $\mathbf{n}$ is i.i.d. zero-mean Gaussian noise. Two different noise levels with standard deviations of 1% and 5%, and five different signals were considered. The considered signals are shown in Fig. 5.7.

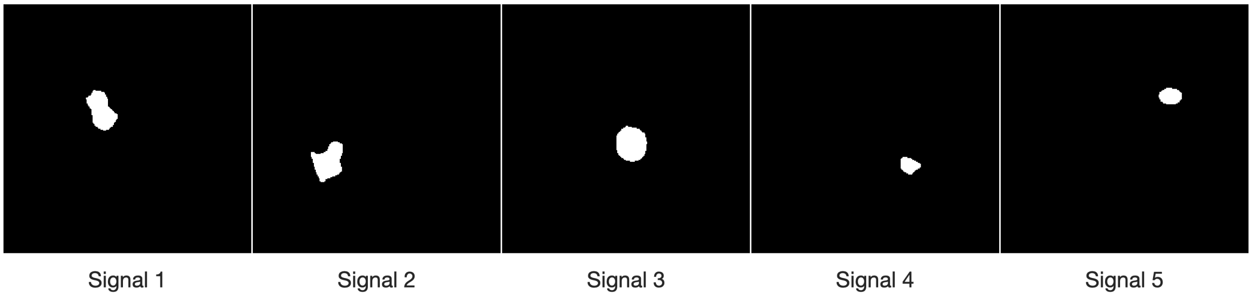


Figure 5.7: Five signals considered in the signal detection study.

Each considered signal detection task was performed on a region of interest (ROI) of dimension of 50 × 50 pixels centered at the signal location. The signal-to-noise ratio of the Hotelling observer (HO) test statistic SNR_{HO} was employed as the figure-of-merit for assessing the image quality [9]:

$$\mathrm{SNR}_{HO} = \sqrt{\mathbf{s}_{ROI}{}^{T}\mathbf{K}^{-1}\mathbf{s}_{ROI}}, \tag{5.4}$$

where $\mathbf{s}_{ROI} \in \mathbb{R}^{2500\times 1}$ denotes the vectorized signal image in the ROI, and $\mathbf{K} \in \mathbb{R}^{2500\times 2500}$ denotes the covariance matrix corresponding to the ROIs in the noisy MR images. When computing SNR_{HO}, $\mathbf{K}^{-1}$ was calculated by use of a covariance matrix decomposition [9]. The values of SNR_{HO} computed by use of 3000 ground truth objects and 3000 generated objects were compared.

5.4.6 Training details

All ProAmGANs and ProGANs were trained by use of Tensorflow [1] by use of 4 NVIDIA Tesla V100 GPUs. The Adam algorithm [63], which is a stochastic gradient algorithm, was employed as the optimizer in the training process. The ProAmGANs were implemented by modifying the ProGAN code (`https://github.com/tkarras/progressive_growing_of_gans`) according to the proposed ProAmGAN architecture illustrated in Fig. 5.2. Specifically, for each considered imaging system, the corresponding measurement operator and the reconstruction operator were applied to the generator-produced images, and the output images were subsequently employed by the discriminator. The training of all ProAmGANs and ProGANs started with a resolution of 4×4. During the training process, the resolution was doubled by gradually adding more layers to the generator and the discriminator until the final resolution was achieved. More details regarding the progressive training details can be found in the literature [59].

5.5 Results

5.5.1 Visual assessments

The ground-truth (top row) and ProAmGAN-generated objects (bottom row) corresponding to chest X-ray images are shown in Fig. 5.8. The ProAmGAN-generated objects have similar visual appearances to the ground-truth ones.

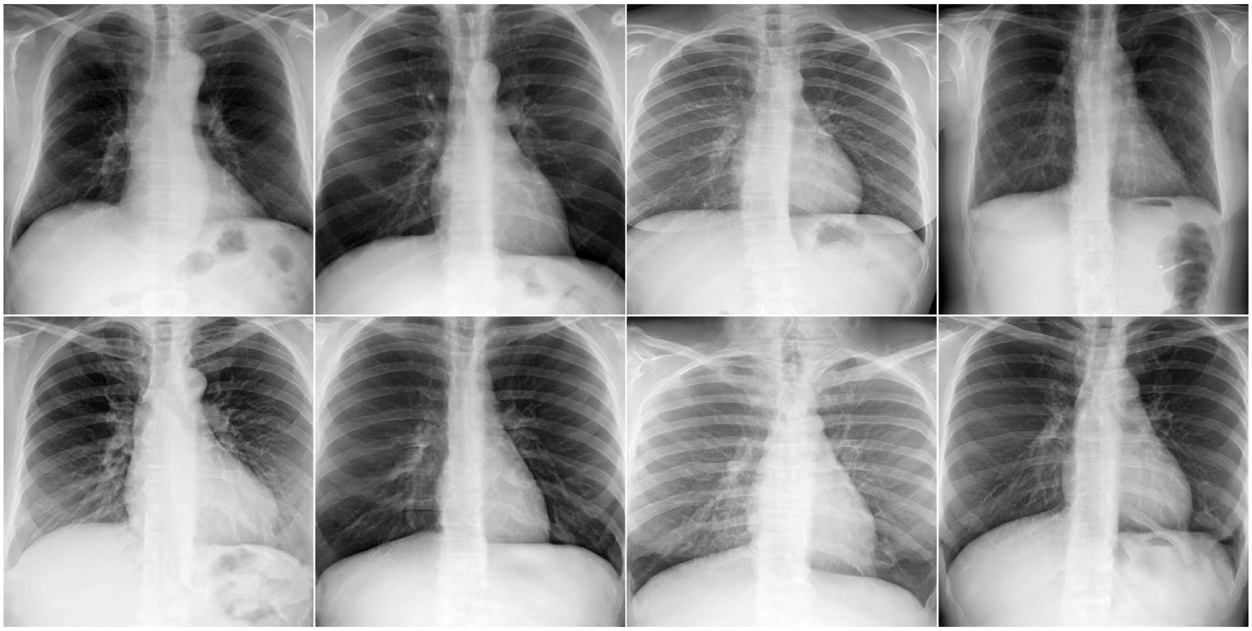
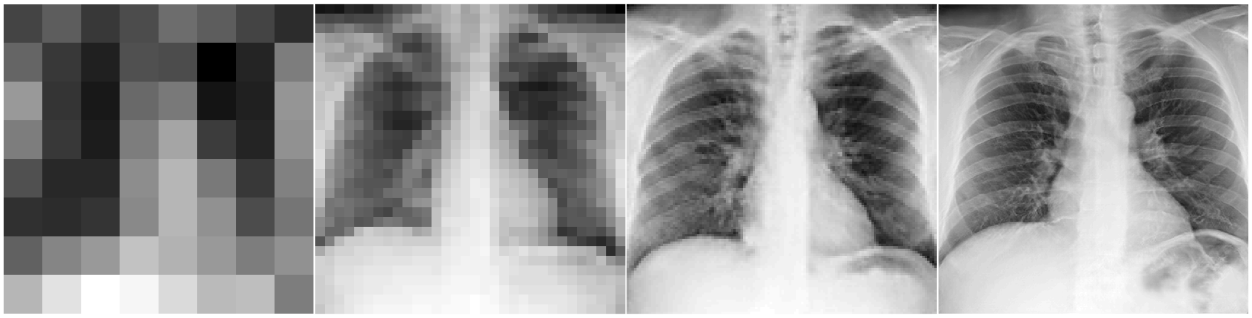


Figure 5.8: Top: Ground-truth chest X-ray objects $\mathbf{f}$. Bottom: ProAmGAN-generated chest X-ray objects $\hat{\mathbf{f}}$.

Synthetic images produced by the ProAmGAN at different training steps corresponding to different image resolutions are shown in Fig. 5.9. FID scores corresponding to different image resolutions were computed.



8×8: FID = 499.2887 32×32: FID = 311.6655 128×128: FID = 114.1847 512×512: FID = 28.7975

Figure 5.9: ProAmGAN-generated chest X-ray images at different training steps. FID scores decreased as the resolution increased in the training process.

A ProGAN-generated and ProAmGAN-generated objects are further compared in Fig. 5.10. It is clear that the ProAmGAN-produced chest X-ray image contains less noise than the one produced by the ProGAN. This demonstrates the ability of the ProAmGAN to mitigate measurement noise when establishing SOMs.

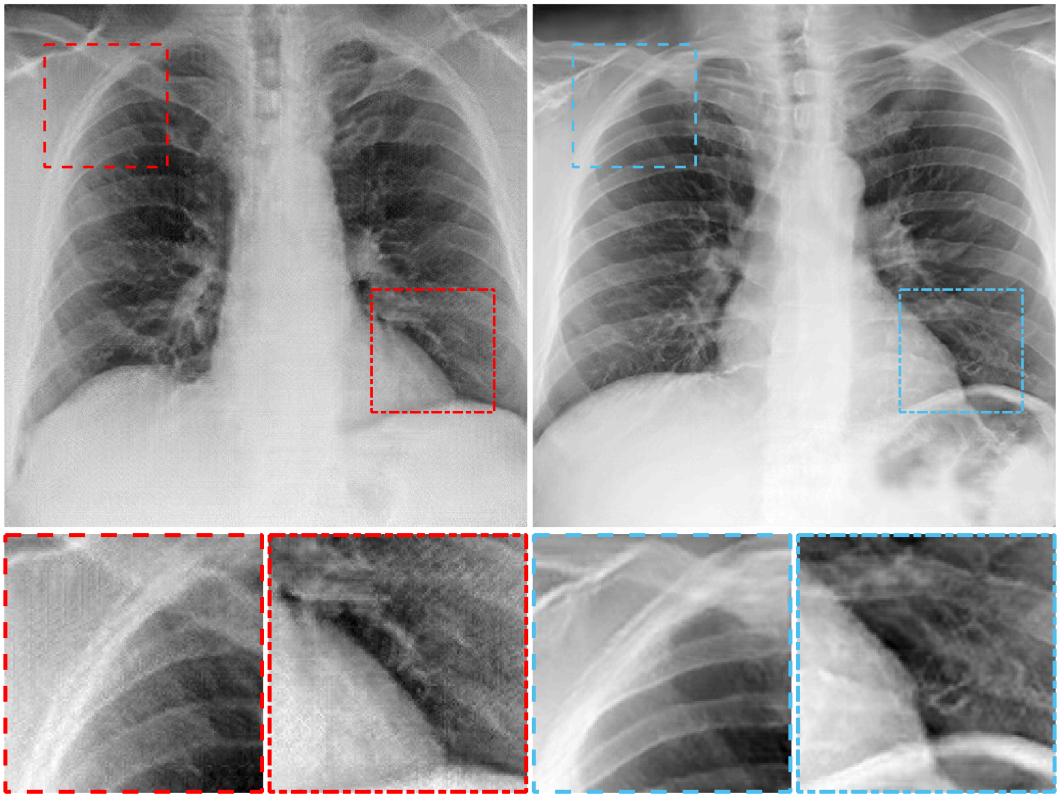


Figure 5.10: A ProGAN-generated (left panel) and ProAmGAN-generated (right panel) chest X-ray object.

The ground-truth (top row) and ProAmGAN-generated objects (bottom row) corresponding to chest CT images are shown in Fig. 5.11. The ProAmGAN-generated objects have similar visual appearances to ground-truth ones.

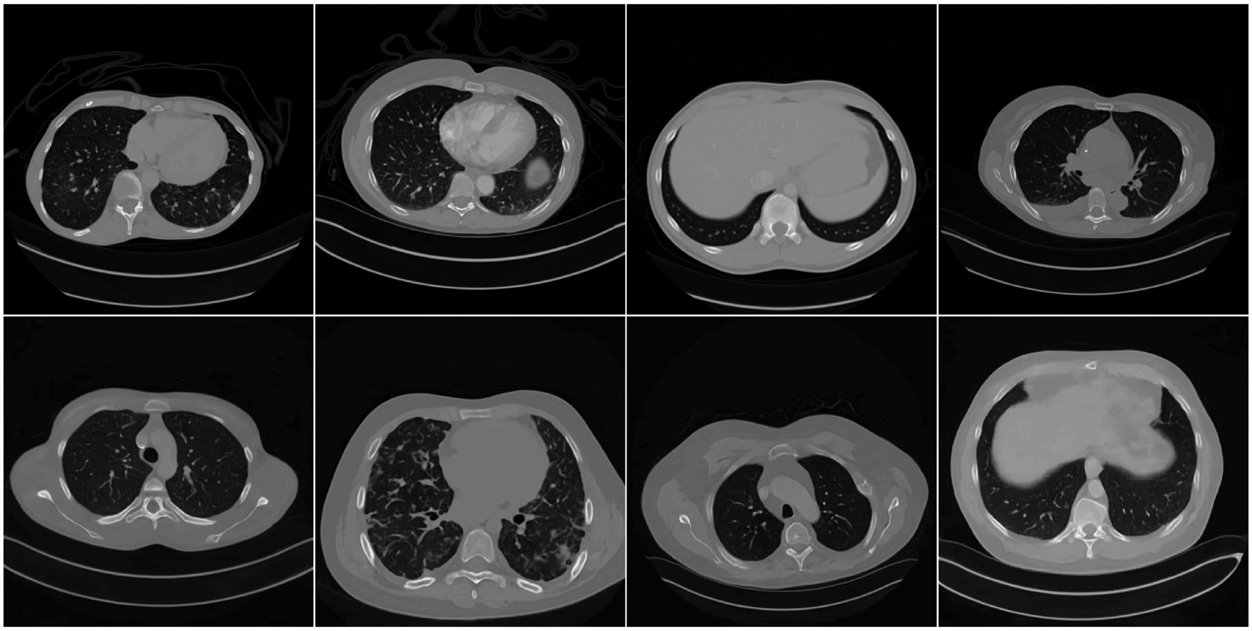
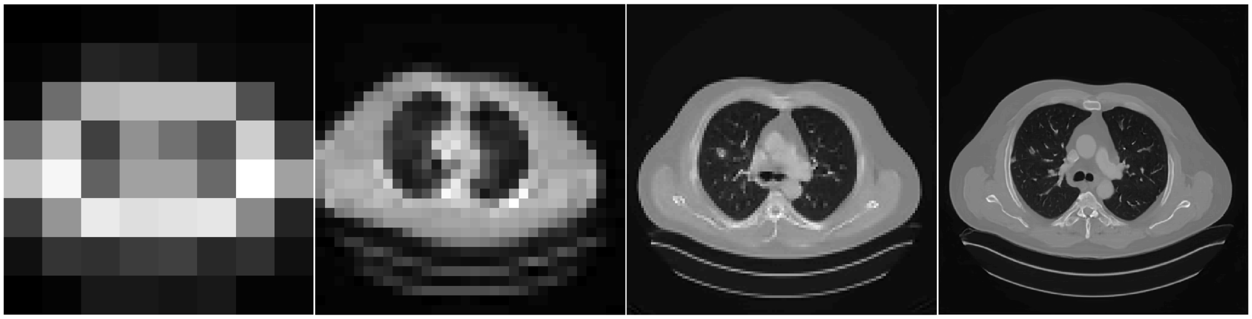


Figure 5.11: Top: Ground-truth chest CT objects $\mathbf{f}$. Bottom: ProAmGAN-generated chest CT objects $\hat{\mathbf{f}}$.

Synthetic chest CT images produced by the ProAmGAN at different training steps corresponding to different image resolutions are shown in Fig. 5.12. FID scores corresponding to different image resolutions were computed.



8×8: FID = 481.2382 32×32: FID = 282.6595 128×128: FID = 165.1643 512×512: FID = 30.6161

Figure 5.12: ProAmGAN-generated chest CT images at different training steps. FID scores decreased as the resolution increased in the training process.

ProGAN-generated and ProAmGAN-generated chest CT images are shown in more detail in Fig. 5.13. It is clear that the ProAmGAN-produced chest CT image in Fig. 5.13 contains fewer artifacts than the one produced by the ProGAN. This demonstrates the ability of the ProAmGAN to mitigate reconstruction artifacts when establishing SOMs.

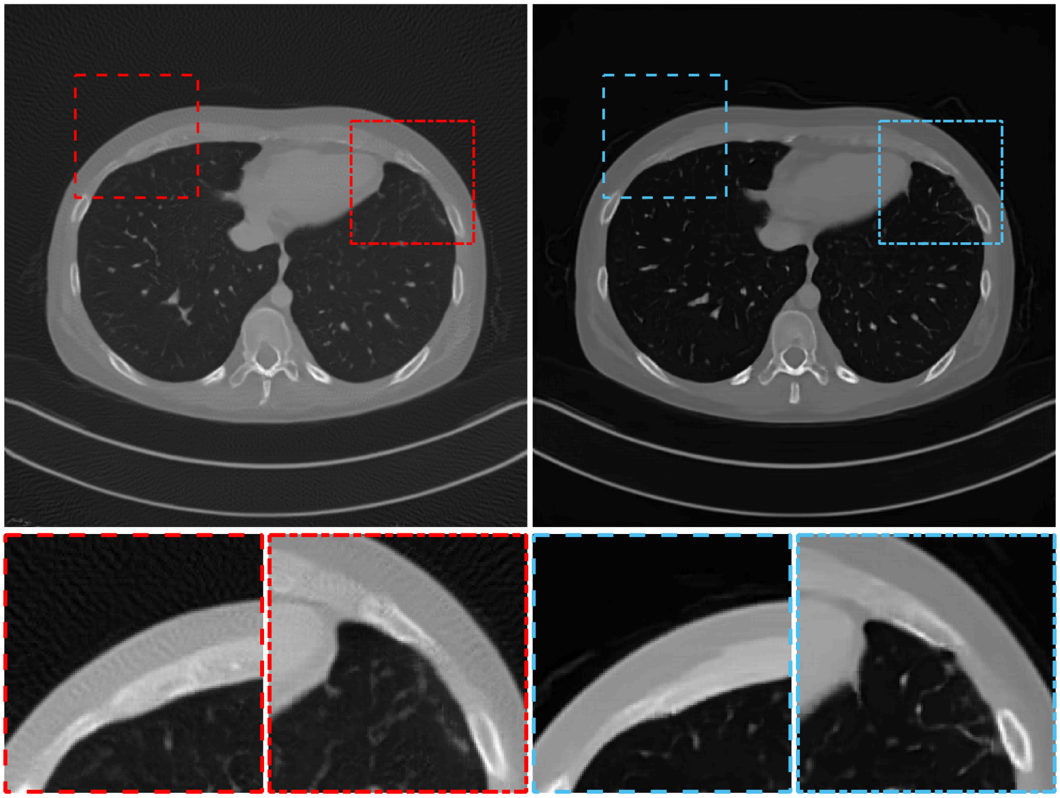


Figure 5.13: A ProGAN-generated (left panel) and ProAmGAN-generated (right panel) chest CT object.

The ground-truth (top row) and ProAmGAN-generated objects (bottom row) corresponding to brain MR images are shown in Figs. 5.14. The ProAmGAN-generated objects have similar visual appearances to ground-truth ones.

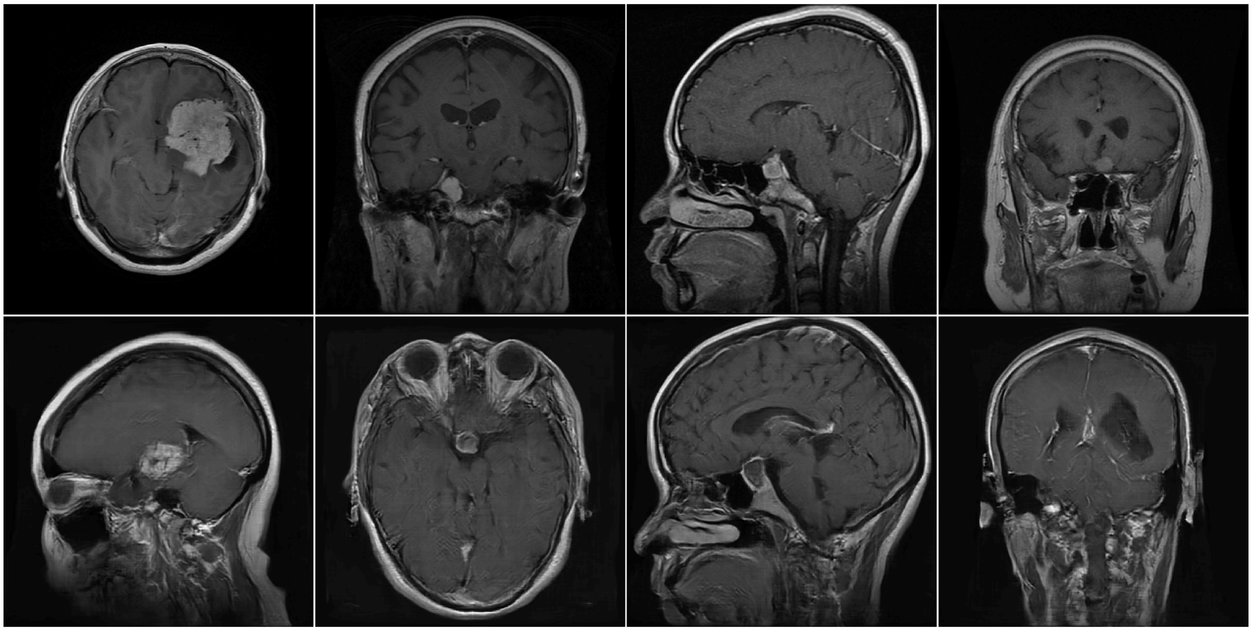
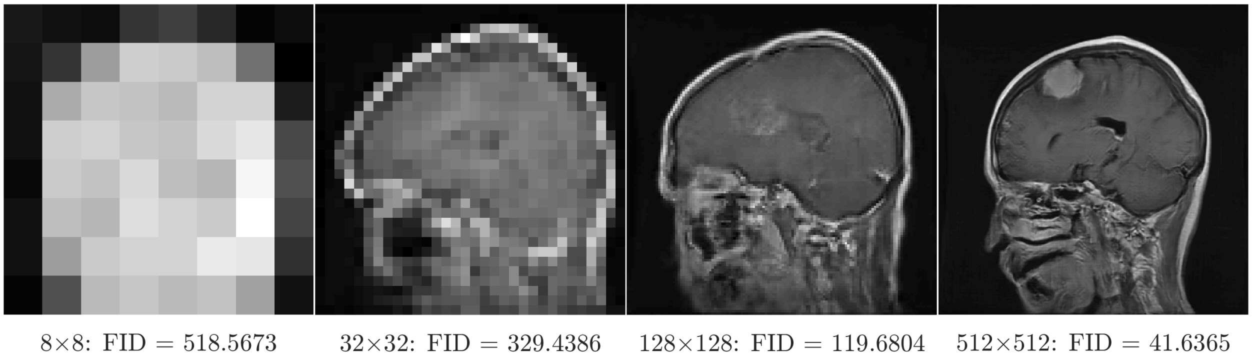


Figure 5.14: Top: Ground-truth brain MR objects $\mathbf{f}$. Bottom: ProAmGAN-generated brain MR objects $\hat{\mathbf{f}}$.

Synthetic brain MR images produced by the ProAmGAN at different training steps corresponding to different image resolutions are shown in Fig. 5.15. FID scores corresponding to different image resolutions were computed.



8×8: FID = 518.5673   32×32: FID = 329.4386   128×128: FID = 119.6804   512×512: FID = 41.6365

Figure 5.15: ProAmGAN-generated brain MR images at different training steps. FID scores decreased as the resolution increased in the training process.

ProGAN-generated and ProAmGAN-generated brain MR images are shown in more detail in Fig. 5.16. The ProAmGAN-produced brain MR image in Fig. 5.16 contains less noise than the one produced by the ProGAN. This demonstrates the ability of the ProAmGAN to mitigate the noise in the reconstructed images when establishing SOMs.

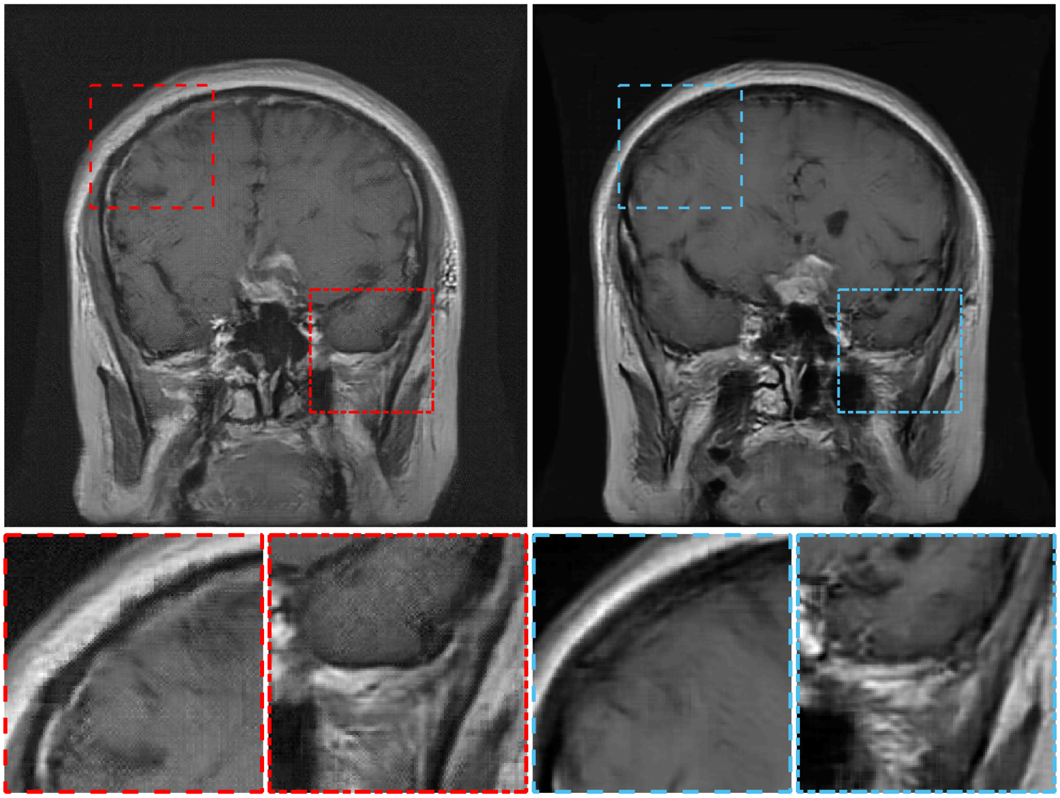


Figure 5.16: A ProGAN-generated (left panel) and ProAmGAN-generated (right panel) brain MR object.

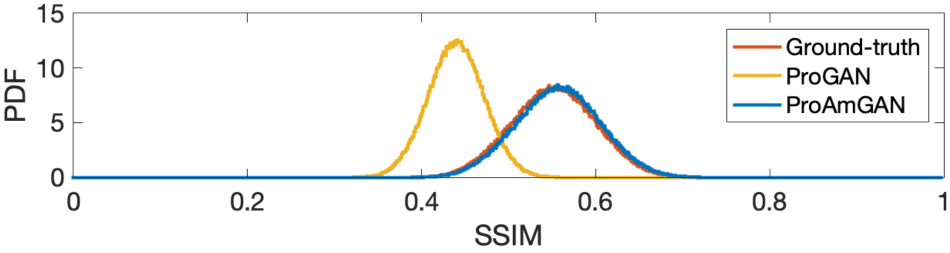
5.5.2 Quantitative assessments

The FID scores corresponding to ProGANs and ProAmGANs for the idealized direct imaging system, computed tomographic imaging system and MR imaging system with complete k-space data are shown in Table 5.2. The ProAmGANs had smaller FID scores than the ProGANs, which indicates that the ProAmGANs outperformed the ProGANs.

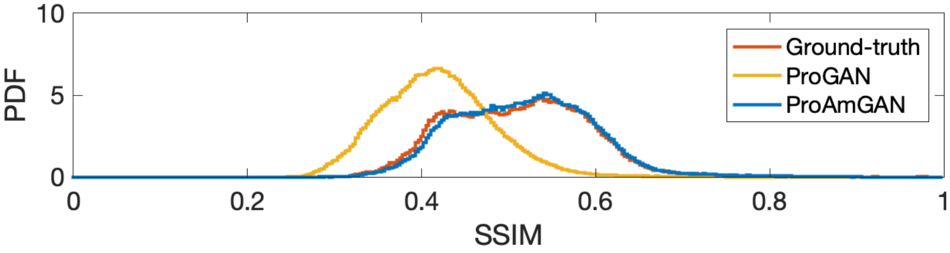
	ProGAN			ProAmGAN		
	X-ray	CT	MRI	X-ray	CT	MRI
FID score	65.5830	62.3854	47.2472	28.7975	30.6161	41.6365
SSIM PDF overlap area	0.1635	0.5230	0.7208	0.9570	0.9599	0.9804
Two-sample KS test statistic	0.8365	0.4772	0.2793	0.0429	0.0384	0.0173

Table 5.2: FID and metrics that evaluate PDFs of SSIMs. Here, "X-ray", "CT", and "MRI" correspond to the idealized direct imaging system, computed tomographic imaging system and MR imaging system with complete k-space data, respectively.

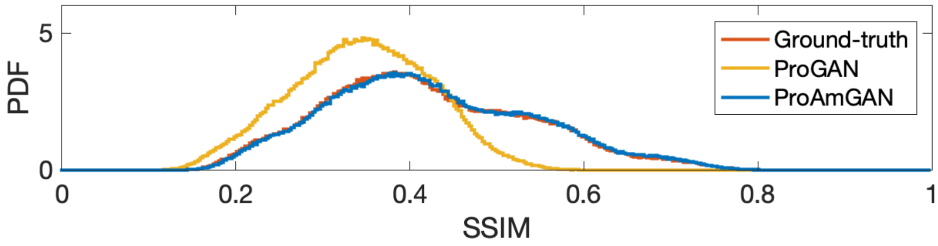
The empirical PDFs of SSIMs corresponding to the idealized direct imaging system, computed tomographic imaging system and MR imaging system with complete k-space data are shown in Fig. 5.17, and the corresponding PDF overlap areas and two-sample KS test statistics are summarized in Table 5.2. The PDFs of SSIMs corresponding to the ProAmGAN-generated and ground-truth objects largely overlap, while the one corresponding to the ProGAN-generated images had a significant discrepancy to the ground-truth PDF.



(a) Idealized direct imaging system



(b) Computed tomographic imaging system



(c) MR imaging system with complete k-space data

Figure 5.17: Empirical PDFs of SSIMs corresponding to ground-truth image pairs (red curves), ground-truth and ProAmGAN-generated image pairs (blue curves), and ground-truth and ProGAN-generated image pairs (yellow curves).

5.5.3 MR imaging system with under-sampled k-space data

The ground-truth (top row) objects and ProAmGAN-generated objects trained with 4/5 k-space sampling ratio (bottom row) are shown in Fig. 5.18. The ProAmGAN-generated objects have similar visual appearances to the ground-truth objects.

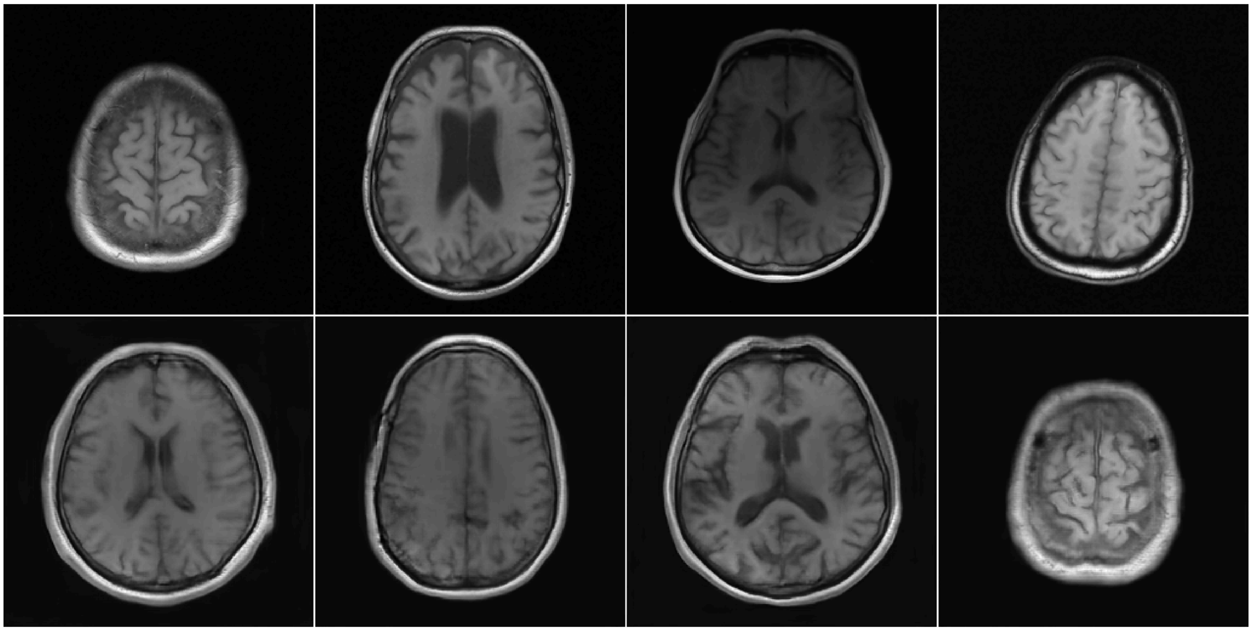


Figure 5.18: Top: Examples of ground-truth objects $\mathbf{f}$. Bottom: Examples of ProAmGAN-generated objects $\hat{\mathbf{f}}$ corresponding to the data-acquisition design with 4/5 k-space sampling ratio.

Objects produced by ProAmGANs and ProGANs trained with different data-acquisition designs are shown in Fig. 5.19. It was observed that the ProAmGAN-generated objects (top row) are visually plausible for the k-space sampling ratios that range from 1/2 to 1/1, while the noise and aliasing artifacts appear in the ProGAN-generated objects (bottom row).

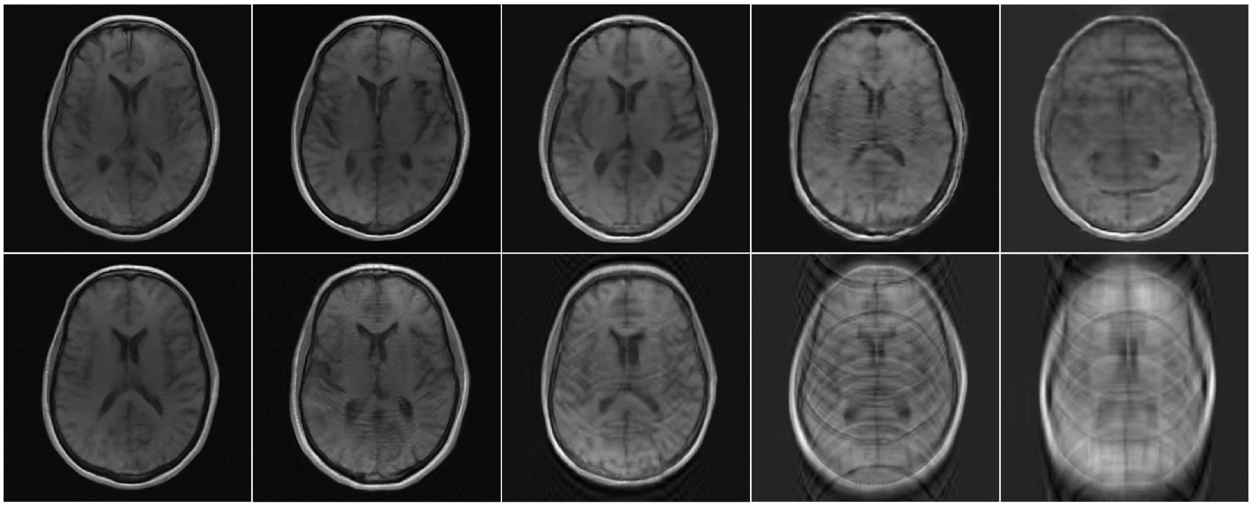


Figure 5.19: ProAmGAN-generated objects (top row) and ProGAN-generated objects (bottom row). From left to right, the ProGAN and ProAmGAN trained with the k-space sampling ratio of 1/1, 4/5, 1/2, 1/4, and 1/8.

The FID corresponding to the objects $\mathbf{f}$ and that corresponding to the measurement components $\mathbf{f}_{meas}$ were computed for evaluating the ProAmGAN that was trained with each data-acquisition design. These FID scores are summarized in Table 5.3. It is observed that the FID between $\mathbf{f}$ and $\hat{\mathbf{f}}$ increased when the k-space sampling ratio decreased, while the FID between $\mathbf{f}_{meas}$ and $\hat{\mathbf{f}}_{meas}$ were not significantly changed. This indicates that the ProAmGANs were unable to establish SOMs by use of measurement data that were acquired by imaging systems having a non-trivial null space, while the variation in the measurement components can be learned.

k-space sampling ratio	1/1	4/5	1/2	1/4	1/8
FID between $\mathbf{f}$ and $\hat{\mathbf{f}}$	30.2247	38.5101	65.4784	105.6070	144.3667
FID between $\mathbf{f}_{meas}$ and $\hat{\mathbf{f}}_{meas}$	30.2247	24.0327	20.3832	19.1034	20.1216

Table 5.3: FID scores corresponding to the objects and the measurement components.

5.5.4 Task-based image quality assessment

The Hotelling observer performance was computed according to Eq. (5.4) and is shown in Fig. 5.20. It was observed that SNR_{HO} has a positive bias when the ProAmGAN is trained with imaging systems that acquire under-sampled k-space data. This is because the ProAmGAN was not able to learn the complete object variation when the imaging system has a non-trivial null-space. When the noise level was increased, the object variation became relatively less important in terms of limiting the observer performance, and the positive bias of SNR_{HO} subsequently became less significant. This is consistent with the observation in reference [65].

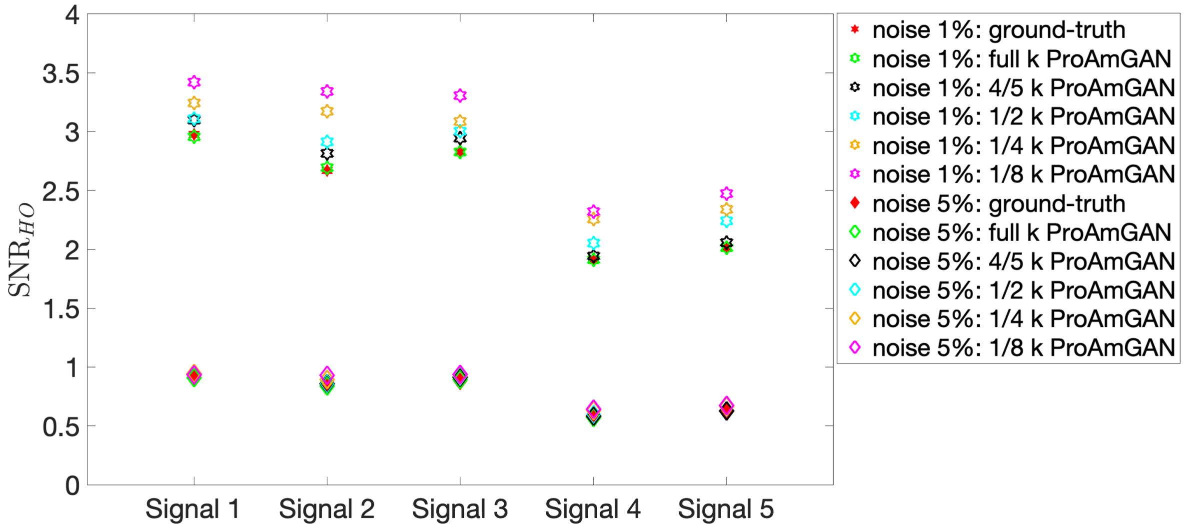


Figure 5.20: Hotelling observer performance corresponding to different tasks with different signals, noise levels, and k-space sampling ratios.

5.6 Discussion and Conclusion

Variation in the objects to-be-imaged can significantly limit the performance of an observer. When conducting computer-simulation studies, this variation can be described by SOMs. In this work, a deep-learning-based method that employed ProAmGANs was developed and investigated for establishing SOMs from measured image data. The proposed ProAmGAN strategy incorporates the advanced progressive growing training procedure and therefore enables the AmbientGAN to be applied to realistically sized medical image data. To demonstrate this, stylized numerical studies were conducted in which ProAmGANs were trained on different object ensembles corresponding to common medical imaging modalities. Both visual examinations and quantitative analyses including task-specific validations indicate that the proposed ProAmGANs hold promise to establish realistic SOMs from imaging measurements.

In addition to objectively assessing imaging systems and data-acquisition designs, the ProAmGAN-established SOMs can be employed to regularize image reconstruction problems. Recent methods have been developed for regularizing image reconstruction problems based on GANs such as Compressed Sensing using Generative Models (CSGM) [14] and image-adaptive GAN-based reconstruction methods (IAGAN) [12, 55]. These methods can be readily employed with the SOMs established by use of the proposed ProAmGANs. ProAmGANs can also be used to produce clean reference images for training deep neural networks for solving other image-processing problems such as image denoising [114] and image super-resolution [33].

It is desirable to establish three-dimensional (3D) object models. A preliminary study developed a progressive-growing 3D GAN [35] and demonstrated its ability to generate 3D MR brain images with the dimension of $64 \times 64 \times 64$. Our proposed method can be readily extended to establish 3D object models by adopting such 3D GAN training strategies. Establishing a 3D version of the ProAmGAN will be explored in the future.

There remain additional topics for future investigation. It is critical to validate the learned SOMs for specific diagnostic tasks. We have conducted preliminary task-specific validation studies by use of the Hotelling Observer [9, 125] and simple binary signal detection tasks. It will be important to validate the learned SOMs for more complicated tasks by use of other observers such as the ideal observer [117, 118, 119, 123] and anthropomorphic observers [75]. Finally, our proposed method can be readily employed with other GAN architectures such as the style-based generator architecture (StyleGAN) [60, 61] that can provide the additional ability to control certain features of generated-images and potentially can further improve the quality of generated-images.

Chapter 6

Markov-Chain Monte Carlo Approximation of the Ideal Observer using Generative Adversarial Networks

6.1 Introduction

As discussed in Chapters 1 and 3. the Ideal Observer (IO) performance has been advocated for use in computing a figure-of-merit (FOM) for assessing and optimizing medical imaging systems. In this way, imaging systems can be optimized in such a way that the amount of task-specific information in the measurement data is maximized. However, the IO test statistic implements the likelihood ratio that is intractable to analytically compute in the majority of cases. To address this difficulty, a sampling-based method that employs Markov-Chain

Monte Carlo (MCMC) techniques [68] was proposed. However, current applications of this method have been limited to some relatively simple stochastic object models (SOMs) such as a lumpy background model [67], a binary texture model [2], and a parameterized torso phantom [50]. It remains unclear how the MCMC techniques can be implemented with other more sophisticated object models.

In this chapter, inspired by the MCMC algorithm developed by Kupinski *et al.* [68], we propose a novel methodology called MCMC-GAN for approximating the IO that implements MCMC techniques with SOMs established by use of GANs. Because the implementation of GANs is general and not limited to specific objects, the proposed MCMC-GAN method can be implemented with sophisticated object models that can be trained by use of GANs and therefore the domain of applicability of MCMC methods can be extended. In numerical studies, binary signal detection tasks that involve clinical brain magnetic resonance (MR) images and clinical brain positron emission tomography (PET) images are considered. Receiver operating characteristic (ROC) curves and the area under the ROC curve (AUC) values corresponding to the proposed MCMC-GAN algorithm are compared to those corresponding to the CNN-approximated IO described in Chapter 3.

6.2 Markov-Chain Monte Carlo method for approximating the IO

Consider a binary signal detection task that requires an observer to classify an image $\mathbf{g}$ as satisfying a signal-absent hypothesis (H_0) or a signal-present hypothesis (H_1). The imaging processes can be represented as:

$$H_0 : \mathbf{g} = \mathbf{b} + \mathbf{n}, \tag{6.1a}$$

$$H_1 : \mathbf{g} = \mathbf{b} + \mathbf{s} + \mathbf{n}, \tag{6.1b}$$

where $\mathbf{b} \in \mathbb{R}^M$ denotes an image of background, $\mathbf{s} \in \mathbb{R}^M$ denotes the signal to be detected, and $\mathbf{n} \in \mathbb{R}^M$ denotes the random measurement noise.

As introduced in Chapter 2, the Ideal Observer (IO) sets an upper performance limit among all observers, and the IO test statistic can be computed as any monotonic transformation of the likelihood ratio:

$$\Lambda(\mathbf{g}) = \frac{p(\mathbf{g}|H_1)}{p(\mathbf{g}|H_0)}. \tag{6.2}$$

However, computation of $\Lambda(\mathbf{g})$ generally is intractable analytically.

Kupinski *et al.* proposed a method to numerically approximate the IO test statistic by employing MCMC techniques [68]. For a signal-known-exactly (SKE) binary signal detection task, the likelihood ratio can be written as [68]:

$$\Lambda(\mathbf{g}) = \frac{\int d\mathbf{b}\ p_b(\mathbf{b}) p(\mathbf{g}|\mathbf{b}, H_1)}{\int d\mathbf{b}\ p_b(\mathbf{b}) p(\mathbf{g}|\mathbf{b}, H_0)} \equiv \int d\mathbf{b}\ \Lambda_{\mathrm{BKE}}(\mathbf{g}|\mathbf{b}) p(\mathbf{b}|\mathbf{g}, H_0), \tag{6.3}$$

where $\Lambda_{\mathrm{BKE}}(\mathbf{g}|\mathbf{b}) = \frac{p(\mathbf{g}|\mathbf{b},H_1)}{p(\mathbf{g}|\mathbf{b},H_0)}$ and $p(\mathbf{b}|\mathbf{g}, H_0) = \frac{p(\mathbf{g}|\mathbf{b},H_0)p_b(\mathbf{b})}{\int d\mathbf{b}' p(\mathbf{g}|\mathbf{b}',H_0)p_b(\mathbf{b}')}$. The BKE likelihood ratio $\Lambda_{\mathrm{BKE}}(\mathbf{g}|\mathbf{b})$ sometimes has an analytical form that is dependent on the type of measurement noise [67]. In cases where the background can be described by a stochastic object model (SOM) with a set of stochastic parameters $\boldsymbol{\theta}$, i.e., $\mathbf{b} \equiv \mathbf{b}(\boldsymbol{\theta})$, the likelihood ratio described in Eq. (6.3) can be written as [68]: $\Lambda(\mathbf{g}) = \int d\boldsymbol{\theta}\ \Lambda_{\mathrm{BKE}}(\mathbf{g}|\mathbf{b}(\boldsymbol{\theta}))p(\boldsymbol{\theta}|\mathbf{g}, H_0)$. Subsequently, the likelihood ratio can be approximated as:

$$\hat{\Lambda}(\mathbf{g}) = \frac{1}{J}\sum_{j=1}^{J}\Lambda_{\mathrm{BKE}}(\mathbf{g}|\mathbf{b}(\boldsymbol{\theta}^j)). \tag{6.4}$$

Here, each $\boldsymbol{\theta}^j$ is sampled from the posterior distribution $p(\boldsymbol{\theta}|\mathbf{g}, H_0)$. To sample $\boldsymbol{\theta}^j$ from the distribution $p(\boldsymbol{\theta}|\mathbf{g}, H_0)$, a Markov chain with the stationary density $p(\boldsymbol{\theta}|\mathbf{g}, H_0)$ can be generated. To do this, an initial vector $\boldsymbol{\theta}^0$ is chosen and a proposal density function $q(\boldsymbol{\theta}|\boldsymbol{\theta}^j)$ is specified. Given $\boldsymbol{\theta}^j$, the candidate vector $\hat{\boldsymbol{\theta}}$ is sampled from the proposal density $q(\boldsymbol{\theta}|\boldsymbol{\theta}^j)$ and it is accepted with probability [68]:

$$p_{\mathrm{a}}(\widehat{\boldsymbol{\theta}}|\boldsymbol{\theta}^j, \mathbf{g}) = \min\left[1, \frac{p(\mathbf{g}|\mathbf{b}(\hat{\boldsymbol{\theta}}), H_0)p(\hat{\boldsymbol{\theta}})q(\boldsymbol{\theta}^j|\hat{\boldsymbol{\theta}})}{p(\mathbf{g}|\mathbf{b}(\boldsymbol{\theta}^j), H_0)p(\boldsymbol{\theta}^j)q(\hat{\boldsymbol{\theta}}|\boldsymbol{\theta}^j)}\right]. \tag{6.5}$$

The vector $\boldsymbol{\theta}^{j+1} \equiv \hat{\boldsymbol{\theta}}$ if the candidate is accepted; otherwise $\boldsymbol{\theta}^{j+1} \equiv \boldsymbol{\theta}^j$. If the proposal density is designed to be symmetric, i.e., $q(\hat{\boldsymbol{\theta}}|\boldsymbol{\theta}^j) = q(\boldsymbol{\theta}^j|\hat{\boldsymbol{\theta}})$, the sampling strategy described above becomes a Metropolis-Hastings approach and the factors corresponding to the proposal density are cancelled.

Park *et al.* extended the MCMC approach to signal-known-statistically (SKS) signal detection tasks [85] where the signal $\mathbf{s}$ is random. If the signal can be described by a set of stochastic parameters $\boldsymbol{\alpha}$, i.e., $\mathbf{s} = \mathbf{s}(\boldsymbol{\alpha})$, the likelihood ratio $\Lambda(\mathbf{g})$ can be written as [85] :

$$\Lambda(\mathbf{g}) = \int d\boldsymbol{\alpha} \int d\boldsymbol{\theta}\ \Lambda_{\mathrm{BSKE}}(\mathbf{g}|\mathbf{b}(\boldsymbol{\theta}), \mathbf{s}(\boldsymbol{\alpha}))p(\boldsymbol{\theta}|\mathbf{g}, H_0)p(\boldsymbol{\alpha}), \tag{6.6}$$

where $\Lambda_{\mathrm{BSKE}}(\mathbf{g}|\mathbf{b}(\boldsymbol{\theta}), \mathbf{s}(\boldsymbol{\alpha})) = \frac{p(\mathbf{g}|\mathbf{b}(\boldsymbol{\theta}), \mathbf{s}(\boldsymbol{\alpha}), H_1)}{p(\mathbf{g}|\mathbf{b}(\boldsymbol{\theta}), H_0)}$. The likelihood ratio can be subsequently approximated as:

$$\hat{\Lambda}(\mathbf{g}) = \frac{1}{J}\sum_{j=1}^{J}\Lambda_{\mathrm{BSKE}}(\mathbf{g}|\mathbf{b}(\boldsymbol{\theta}^j), \mathbf{s}(\boldsymbol{\alpha}^j)). \tag{6.7}$$

Here, $(\boldsymbol{\theta}^j, \boldsymbol{\alpha}^j)$ are sampled from the distribution $p(\boldsymbol{\theta}|\mathbf{g}, H_0)p(\boldsymbol{\alpha})$. The Markov chain can be constructed with acceptance probability:

$$p_{\mathrm{a}}(\widehat{\boldsymbol{\theta}}, \widehat{\boldsymbol{\alpha}}|\boldsymbol{\theta}^j, \boldsymbol{\alpha}^j, \mathbf{g}) = \min\left[1, \frac{p(\mathbf{g}|\mathbf{b}(\hat{\boldsymbol{\theta}}), H_0)p(\hat{\boldsymbol{\theta}})p(\hat{\boldsymbol{\alpha}})q(\boldsymbol{\theta}^j|\hat{\boldsymbol{\theta}})q(\boldsymbol{\alpha}^j|\hat{\boldsymbol{\alpha}})}{p(\mathbf{g}|\mathbf{b}(\boldsymbol{\theta}^j), H_0)p(\boldsymbol{\theta}^j)p(\boldsymbol{\alpha}^j)q(\hat{\boldsymbol{\theta}}|\boldsymbol{\theta}^j)q(\hat{\boldsymbol{\alpha}}|\boldsymbol{\alpha}^j)}\right]. \tag{6.8}$$

Again, if the proposal densities are designed to be symmetric, the factors corresponding to the proposal density in Eq. (6.8) are canceled.

However, implementations of these MCMC methods can be difficult due to practical issues such as the design of proposal density for the considered object model. In addition, it remains unclear how to apply these methods for situations where the background cannot be described by well-established SOMs.

6.3 Markov-Chain Monte Carlo approximation of the IO by use of GANs

As introduced in Chapter 5, deep generative neural networks such as generative adversarial networks (GANs) [46] hold great potential to learn statistical properties of training images and generate new images that consistent with them. Once a GAN has been trained on a set of background images $\mathbf{b}$, the generator can be employed to generate synthesized background images $\hat{\mathbf{b}}$: $\hat{\mathbf{b}} = G(\mathbf{z}; \boldsymbol{\Theta}_G)$. Here, $G(\cdot\ \boldsymbol{\Theta}_G) : \mathbb{R}^k \to \mathbb{R}^M$ is a mapping function represented by a deep neural network with a weight vector $\boldsymbol{\Theta}_G$, and $\mathbf{z} \in \mathbb{R}^k$ is a latent vector that is sampled from a known distribution such as normal distribution. The probability distribution of the real background images p_b can be subsequently approximated by the probability distribution of the GAN-produced background images $p_{\hat{b}}$.

The IO test statistic for SKE binary signal detection tasks can be subsequently approximated as:

$$\Lambda(\mathbf{g}) = \frac{\int d\hat{\mathbf{b}}\ p_{\hat{b}}(\hat{\mathbf{b}})p(\mathbf{g}|\hat{\mathbf{b}}, H_1)}{\int d\hat{\mathbf{b}}\ p_{\hat{b}}(\hat{\mathbf{b}})p(\mathbf{g}|\hat{\mathbf{b}}, H_0)} \equiv \int d\hat{\mathbf{b}}\ \Lambda_{\mathrm{BKE}}(\mathbf{g}|\hat{\mathbf{b}})p(\hat{\mathbf{b}}|\mathbf{g}, H_0), \tag{6.9}$$

where $\Lambda_{\mathrm{BKE}}(\mathbf{g}|\hat{\mathbf{b}}) = \frac{p(\mathbf{g}|\hat{\mathbf{b}},H_1)}{p(\mathbf{g}|\hat{\mathbf{b}},H_0)}$ and $p(\hat{\mathbf{b}}|\mathbf{g}, H_0) = p(\mathbf{g}|\hat{\mathbf{b}}, H_0)p_{\hat{b}}(\hat{\mathbf{b}})/\int d\hat{\mathbf{b}}'\ p(\mathbf{g}|\hat{\mathbf{b}}', H_0)p_{\hat{b}}(\hat{\mathbf{b}}')$. Because $p(\hat{\mathbf{b}}|\mathbf{g}, H_0) = \int d\mathbf{z}\ \delta(\hat{\mathbf{b}} - G(\mathbf{z}; \boldsymbol{\Theta}_G))p(\mathbf{z}|\mathbf{g}, H_0)$, where $\delta(\cdot)$ is a Dirac delta function and $p(\mathbf{z}|\mathbf{g}, H_0) = \frac{p(\mathbf{g}|G(\mathbf{z};\boldsymbol{\Theta}_G),H_0)p_z(\mathbf{z})}{\int d\mathbf{z}' p(\mathbf{g}|G(\mathbf{z}';\boldsymbol{\Theta}_G),H_0)p_z(\mathbf{z}')}$, the likelihood ratio can be rewritten as:

$$\begin{aligned}\Lambda(\mathbf{g}) &= \int d\hat{\mathbf{b}} \int d\mathbf{z}\ \Lambda_{\mathrm{BKE}}(\mathbf{g}|\hat{\mathbf{b}})\delta(\hat{\mathbf{b}} - G(\mathbf{z}; \boldsymbol{\Theta}_G))p(\mathbf{z}|\mathbf{g}, H_0) \\ &= \int d\mathbf{z}\ \Lambda_{\mathrm{BKE}}(\mathbf{g}|G(\mathbf{z}; \boldsymbol{\Theta}_G))p(\mathbf{z}|\mathbf{g}, H_0),\end{aligned} \tag{6.10}$$

where $\Lambda_{\mathrm{BKE}}(\mathbf{g}|G(\mathbf{z}; \boldsymbol{\Theta}_G))$ is evaluated on the synthetic background image generated by the GAN. The likelihood ratio subsequently can be approximated as:

$$\hat{\Lambda}(\mathbf{g}) = \frac{1}{J}\sum_{j=1}^{J} \Lambda_{\mathrm{BKE}}(\mathbf{g}|G(\mathbf{z}^j; \boldsymbol{\Theta}_G)), \tag{6.11}$$

where $\mathbf{z}^j$ is sampled from the posterior distribution $p(\mathbf{z}|\mathbf{g}, H_0)$. To produce $\mathbf{z}^j$, a Markov chain can be constructed by specifying a proposal density function $q(\mathbf{z}|\mathbf{z}^j)$. Given the current sample $\mathbf{z}^j$, a candidate latent vector $\hat{\mathbf{z}}$ is drawn from the proposal density function and is accepted to the Markov chain with the acceptance probability:

$$p_{\mathrm{a}}(\widehat{\mathbf{z}}|\mathbf{z}^j, \mathbf{g}) = \min\left[1, \frac{p\big(\mathbf{g}|G(\hat{\mathbf{z}}; \boldsymbol{\Theta}_G), H_0\big)p_z(\hat{\mathbf{z}})q(\mathbf{z}^j|\hat{\mathbf{z}})}{p\big(\mathbf{g}|G(\mathbf{z}^j; \boldsymbol{\Theta}_G), H_0\big)p_z(\mathbf{z}^j)q(\hat{\mathbf{z}}|\mathbf{z}^j)}\right]. \tag{6.12}$$

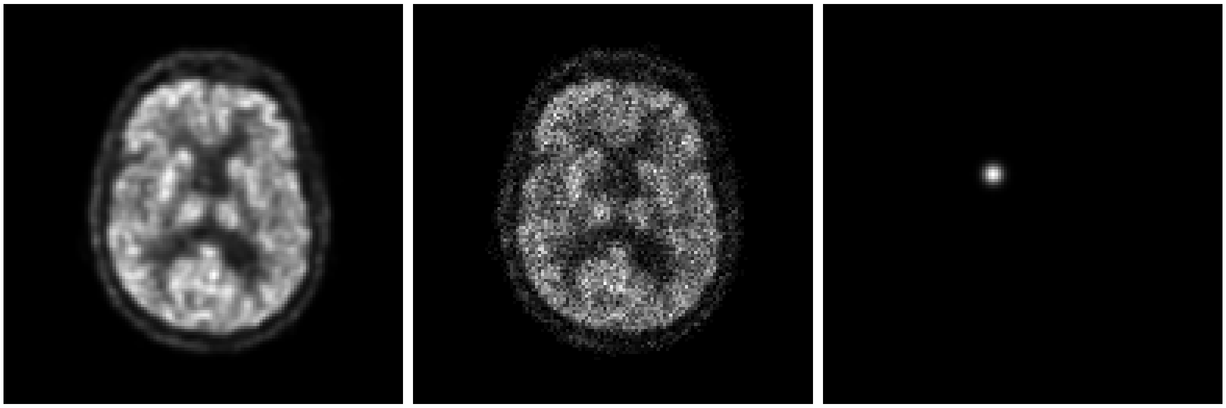
Here, the probability density function $p_z(\cdot)$ has a simple analytical form because the latent vector $\mathbf{z}$ is sampled from a known distribution such as the normal distribution. When a random walk Metropolis-Hastings (RWMH) algorithm [86] is employed, the proposal density $q(\hat{\mathbf{z}}|\mathbf{z}^j)$ can be chosen as a Gaussian density. Additionally, because the gradient of the function represented by the generator $G(\mathbf{z}; \mathbf{\Theta}_G)$ with respect to the latent vector $\mathbf{z}$ can be readily computed, more advanced MH algorithms including Metropolis adjusted Langevin algorithms (MALA) and Hamiltonian Monte Carlo (HMC) [86] that employ gradient information can be employed.

6.4 Numerical studies

Computer-simulation studies were conducted to investigate the ability of the proposed MCMC-GAN method to approximate the IO test statistic associated with SOMs that are established by use of GANs. Two SKE/BKS binary signal detection tasks that involve clinical brain PET images and clinical brain MR images were considered. The observer performance was assessed by use of the ROC curve that was fit by use of the Metz-ROC software [76] that utilized the "proper" binormal model [77, 87]. Details of the computer-simulation studies are provided below.

6.4.1 Clinical brain positron emission tomography (PET) images

A clinical brain PET dataset sponsored by Alzheimer's Disease Neuroimaging Initiative (ADNI) [78] was considered. Eleven thousand high quality images having the dimension of 128×128 were selected to form a dataset for training a GAN for establishing the SOM. These images were subsequently normalized between 0 and 1 for use as training images for training a GAN. After the training, the generator in the trained GAN was employed as a SOM. Poisson noise was employed to simulate the low-dose PET images for use in the considered signal detection task. Specifically, the signal-absent low-dose PET images were generated from the Poisson distribution with the mean $\hat{\mathbf{b}}$ that is the generator-produced images multiplied with 20, and the signal-present low-dose PET images were generated from the Poisson distribution with the mean $\hat{\mathbf{b}} + \mathbf{s}$. Let $G(\mathbf{z}; \mathbf{\Theta}_G)$ denote the function that maps a latent vector $\mathbf{z}$ to the scaled GAN-generated image. The synthesized background image can be represented as: $\hat{\mathbf{b}} = G(\mathbf{z}; \mathbf{\Theta}_G)$. An example of the original PET brain images $\mathbf{b}$ and the corresponding signal-absent low-dose PET image $\mathbf{g}$ are shown in Fig. 6.1 (a) and (b), respectively. The signal image $\mathbf{s}$ corresponding to the considered signal detection task is shown in Fig. 6.1 (c).



(a) (b) (c)

Figure 6.1: (a) An image from the ADNI PET dataset. (b) A simulated low-dose PET image corresponding to (a). (c) The signal image corresponds to the considered signal detection task.

The binary signal detection task was performed on a region of interest (ROI) of dimension of 16 × 16 pixels centered at the signal location. Let $\hat{\mathbf{b}}_{ROI}(\mathbf{z}^j)$ denote the ROI of the GAN-produced background image $\hat{\mathbf{b}}(\mathbf{z}^j) \equiv G(\mathbf{z}^j; \Theta_G)$ and $\mathbf{g}_{\hat{ROI}}$ denote the ROI of the measured image $\hat{\mathbf{g}}$ that corresponds to the GAN-produced image. The IO test statistic can be approximated as:

$$\hat{\Lambda}(\hat{\mathbf{g}}_{ROI}) = \frac{1}{J} \sum_{j=1}^{J} \Lambda_{\mathrm{BKE}}(\hat{\mathbf{g}}_{ROI} | \hat{\mathbf{b}}_{ROI}(\mathbf{z}^j)), \tag{6.13}$$

where $\hat{\mathbf{b}}_{ROI}(\mathbf{z}^j)$ is the ROI of $\hat{\mathbf{b}}(\mathbf{z}^j) \equiv G(\mathbf{z}^j;\Theta_G)$ and $\hat{\mathbf{g}}_{ROI}$ is the ROI of $\hat{\mathbf{g}}$. Because Poisson noise was considered, the BKE likelihood ratio $\Lambda_{\mathrm{BKE}}(\hat{\mathbf{g}}_{ROI}|\hat{\mathbf{b}}_{ROI}(\mathbf{z}^j))$ can be computed as:

$$\Lambda_{\mathrm{BKE}}(\hat{\mathbf{g}}_{ROI}|\hat{\mathbf{b}}_{ROI}(\mathbf{z}^j)) = \prod_{m=1}^{M} \left(1 + \frac{s_m}{b_m}\right)^{g_m} \exp(-s_m), \tag{6.14}$$

where $M = 256$ and b_m, g_m and s_m are respectively the m^{th} element of $\hat{\mathbf{b}}_{ROI}(\mathbf{z}^j)$, $\hat{\mathbf{g}}_{ROI}$ and $\mathbf{s}_{ROI}$. Here, $\mathbf{s}_{ROI}$ denotes the ROI of the signal image $\mathbf{s}$. The latent vector $\mathbf{z}^j$ was drawn from the posterior distribution $p(\mathbf{z}|\hat{\mathbf{g}}_{ROI}, H_0)$:

$$p(\mathbf{z}|\hat{\mathbf{g}}_{ROI}, H_0) \propto p(\hat{\mathbf{g}}_{ROI}|\hat{\mathbf{b}}_{ROI}(\mathbf{z}^j), H_0)p_z(\mathbf{z}). \tag{6.15}$$

Because Poisson noise was considered, the likelihood function $p(\hat{\mathbf{g}}_{ROI}|\hat{\mathbf{b}}_{ROI}(\mathbf{z}^j), H_0)$ can be described as:

$$p(\hat{\mathbf{g}}_{ROI}|\hat{\mathbf{b}}_{ROI}(\mathbf{z}^j), H_0) = \prod_{m=1}^{M} \exp(-b_m)\frac{(b_m)^{g_m}}{g_m!}. \tag{6.16}$$

The probability density function $p_z(\mathbf{z})$ was described by a standard normal distribution because of the specification of the latent vector $\mathbf{z}$ in the GAN training. To construct the Markov chain, the RWMH algorithm with a proposal density function $q(\mathbf{z}|\mathbf{z}^j)$ that was described by a multivariate Gaussian distribution was employed:

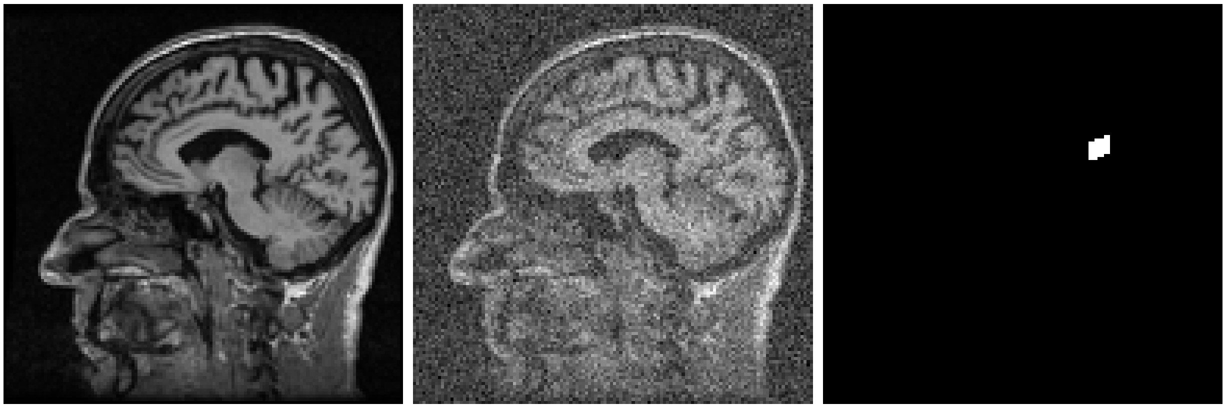
$$q(\hat{\mathbf{z}}|\mathbf{z}^j) \propto \exp\left(-\frac{1}{2\sigma^2}||\hat{\mathbf{z}} - \mathbf{z}^j||_2^2\right). \tag{6.17}$$

Here, the standard deviation σ was set to 0.06. The IO performance was evaluated on 200 signal-absent images and 200 signal-present images. For each image, a Markov chain was constructed by running 500,000 iterations with 5000 burn-in iterations that were discarded.

To validate the proposed MCMC-GAN method, the supervised learning method described in Chapter 3 that employs convolutional neural networks (CNNs) was implemented as a reference method. When training CNNs, a training dataset that comprised one million GAN-generated background images and a "semi-online learning" method in which the measurement noise was generated on-the-fly were employed. A CNN having 13 convolutional (CONV) layers was specified to approximate the IO. Each CONV layer comprised 32 filters with 5 × 5 spatial support and was followed by a LeakyReLU activation function. The last CONV layer was followed by a max-pooling layer and a fully connected (FC) layer. The Hotelling observer (HO) was also computed to provide an additional comparison to the MCMC-GAN approximated IO. The Hotelling template was computed by use of a covariance matrix decomposition in which the background covariance matrix was estimated by use of one million GAN-generated images.

6.4.2 Clinical brain MR images

A clinical brain MR dataset sponsored by ADNI [78] was employed for establishing a SOM by use of GANs. Twelve thousand high quality sagittal brain MR images were selected and resized to the dimension of 128×128. These images were subsequently normalized between 0 and 1 for use as training images for training a GAN. After the training, the generator in the trained GAN was employed as a SOM that describes the variability of the background images $\mathbf{b}$. The noise $\mathbf{n}$ was modeled by independent and identically distributed distributed (i.i.d.) Gaussian random vector with a standard deviation of 0.1. An example of the considered MR brain images $\mathbf{b}$ and the corresponding measured noisy MR images $\mathbf{g}$ is shown in Fig. 6.2 (a) and (b), respectively. The signal image $\mathbf{s}$ for the considered signal detection task is shown in Fig. 6.2 (c).



(a) (b) (c)

Figure 6.2: (a) An image from the ADNI MR dataset. (b) A signal-absent image that is generated by adding i.i.d. Gaussian noise to (a). (c) The signal image corresponds to the considered signal detection task.

A binary signal detection task was performed on a region of interest (ROI) of dimension of 16×16 pixels centered at the signal location. The IO test statistic corresponding to the ROI can be approximated as:

$$\hat{\Lambda}(\hat{\mathbf{g}}_{ROI}) = \frac{1}{J}\sum_{j=1}^{J} \Lambda_{\mathrm{BKE}}(\hat{\mathbf{g}}_{ROI}|\hat{\mathbf{b}}_{ROI}(\mathbf{z}^j)). \tag{6.18}$$

Because i.i.d. Gaussian noise was considered, the BKE likelihood ratio $\Lambda_{\mathrm{BKE}}(\hat{\mathbf{g}}_{ROI}|\hat{\mathbf{b}}_{ROI}(\mathbf{z}^j))$ can be computed as:

$$\Lambda_{\mathrm{BKE}}(\hat{\mathbf{g}}_{ROI}|\hat{\mathbf{b}}_{ROI}(\mathbf{z}^j)) = \exp\left[(\hat{\mathbf{g}}_{ROI} - \hat{\mathbf{b}}_{ROI}(\mathbf{z}^j) - \mathbf{s}_{ROI}/2)^T K_n^{-1}\mathbf{s}_{ROI}\right], \tag{6.19}$$

where K_n is the covariance matrix corresponding to i.i.d. Gaussian noise with standard deviation of 0.1. The latent vector $\mathbf{z}^j$ was drawn from the posterior distribution $p(\mathbf{z}|\hat{\mathbf{g}}_{ROI}, H_0)$:

$$p(\mathbf{z}|\hat{\mathbf{g}}_{ROI}, H_0) \propto p(\hat{\mathbf{g}}_{ROI}|\hat{\mathbf{b}}_{ROI}(\mathbf{z}^j), H_0)p_z(\mathbf{z}). \tag{6.20}$$

The likelihood function $p(\hat{\mathbf{g}}_{ROI}|\hat{\mathbf{b}}_{ROI}(\mathbf{z}^j), H_0)$ for Gaussian noise can be described as:

$$p(\hat{\mathbf{g}}_{ROI}|\hat{\mathbf{b}}_{ROI}(\mathbf{z}^j), H_0) \propto \exp\left[-\frac{1}{2}(\hat{\mathbf{g}}_{ROI} - \hat{\mathbf{b}}_{ROI}(\mathbf{z}^j))^T K_n^{-1}(\hat{\mathbf{g}}_{ROI} - \hat{\mathbf{b}}_{ROI}(\mathbf{z}^j))\right]. \tag{6.21}$$

The probability density function $p_z(\mathbf{z})$ was described by a standard normal distribution because of the specification of the latent vector $\mathbf{z}$ in the GAN training. To construct the Markov chain, the RWMH algorithm with a proposal density function $q(\mathbf{z}|\mathbf{z}^j)$ that was described by a multivariate Gaussian distribution was employed:

$$q(\hat{\mathbf{z}}|\mathbf{z}^j) \propto \exp\left(-\frac{1}{2\sigma^2}||\hat{\mathbf{z}} - \mathbf{z}^j||_2^2\right). \tag{6.22}$$

Here, the standard deviation σ was set to 0.06. The IO performance was evaluated on 200 signal-absent images and 200 signal-present images. For each image, a Markov chain was constructed by running 500,000 iterations with 5000 burn-in iterations that were discarded.

To validate the proposed MCMC-GAN method, the supervised learning method described in Chapter 3 that employs convolutional neural networks (CNNs) was implemented as a reference method. When training CNNs, a training dataset that comprised one million GAN-generated background images and a “semi-online learning” method in which the measurement noise was generated on-the-fly were employed. A CNN having 13 convolutional (CONV) layers was specified to approximate the IO. Each CONV layer comprised 32 filters with 5 × 5 spatial support and was followed by a LeakyReLU activation function. The last CONV layer was followed by a max-pooling layer and a fully connected (FC) layer. The Hotelling observer (HO) was also computed to provide an additional comparison to the MCMC-GAN approximated IO. The Hotelling template was computed by use of a covariance matrix decomposition in which the background covariance matrix was estimated by use of one million GAN-generated images.

6.4.3 GAN training details

Progressively growing GANs (ProGANs) [63] were trained on the considered image datasets to establish SOMs. A latent vector having 64 elements was employed as the input to the generator. More details of the ProGAN architecture used in this study is summarized in Table 6.1. The ProGAN was implemented by use of the ProGAN code (`https://github.com/tkarras/progressive_growing_of_gans`). The ProGANs were trained by use of Tensorflow [1] by use of 4 NVIDIA Quadro RTX 8000 GPUs. The Adam algorithm [63] was employed as the optimizer in the training process.

Generator	Act.	Output shape
Latent vector	-	64×1×1
Conv 4×4	LReLU	64×4×4
Conv 3×3	LReLU	64×4×4
Upscale	-	64×8×8
Conv 3×3	LReLU	64×8×8
Conv 3×3	LReLU	64×8×8
Upscale	-	64×16×16
Conv 3×3	LReLU	64×16×16
Conv 3×3	LReLU	64×16×16
Upscale	-	64×32×32
Conv 3×3	LReLU	64×32×32
Conv 3×3	LReLU	64×32×32
Upscale	-	64×64×64
Conv 3×3	LReLU	64×64×64
Conv 3×3	LReLU	64×64×64
Upscale	-	64×128×128
Conv 3×3	LReLU	64×128×128
Conv 3×3	LReLU	64×128×128
Conv 1×1	linear	1×128×128

Discriminator	Act.	Output shape
Input image	-	1 × 128 × 128
Conv 1 × 1	LReLU	64 × 128 × 128
Conv 3 × 3	LReLU	64 × 128 × 128
Conv 3 × 3	LReLU	64 × 128 × 128
Downscale	-	64 × 64 × 64
Conv 3 × 3	LReLU	64 × 64 × 64
Conv 3 × 3	LReLU	64 × 64 × 64
Downscale	-	64 × 32 × 32
Conv 3 × 3	LReLU	64 × 32 × 32
Conv 3 × 3	LReLU	64 × 32 × 32
Downscale	-	64 × 16 × 16
Conv 3 × 3	LReLU	64 × 16 × 16
Conv 3 × 3	LReLU	64 × 16 × 16
Downscale	-	64 × 8 × 8
Conv 3 × 3	LReLU	64 × 8 × 8
Conv 3 × 3	LReLU	64 × 8 × 8
Downscale	-	64 × 4 × 4
Minibatch stddev	-	65 × 4 × 4
Conv 3 × 3	LReLU	64 × 4 × 4
Conv 4 × 4	LReLU	64 × 1 × 1
Fully-connected	linear	1 × 1 × 1

Table 6.1: The architecture of the generator and discriminator for establishing SOMs corresponding to the considered brain PET and MR images. More details about each component in the architecture can be found in ProGAN paper [59].

6.5 Results

6.5.1 Clinical brain PET images

The GAN-generated images (bottom row) and ground-truth images (top row) are shown in Fig. 6.3. The ProGAN-generated images have similar visual appearances to the ground-truth ones.

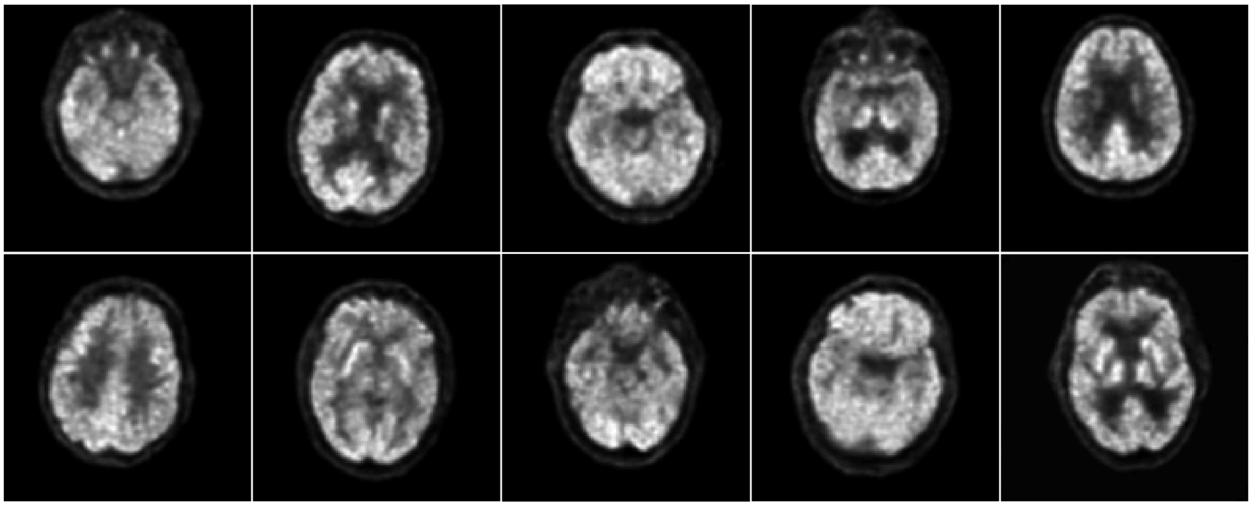


Figure 6.3: Top: Examples of ground-truth PET images. Bottom: Examples of ProGAN-generated PET images.

The ProGAN-generated images have similar visual appearance to the ground-truth images. The Fréchet Inception Distance (FID) [53] score was computed to evaluate the performance of the ProGAN. Lower FID score indicates better quality and diversity of the generated objects. The FID score was 7.7496 that was evaluated on the original 11,000 ground-truth images and 11,000 ProGAN-generated images.

The ROC curves corresponding to the MCMC-GAN IO (blue curve), CNN-IO (red-dashed curve) and the HO (yellow curve) are shown in Fig. 6.6. The curves of the MCMC-GAN IO and CNN-IO are in close agreement, and are higher than the curve of the HO as expected. The AUC value corresponding to the MCMC-GAN IO, CNN-IO and the HO are 0.804 ± 0.021, 0.799 ± 0.022 and 0.618 ± 0.027, respectively.

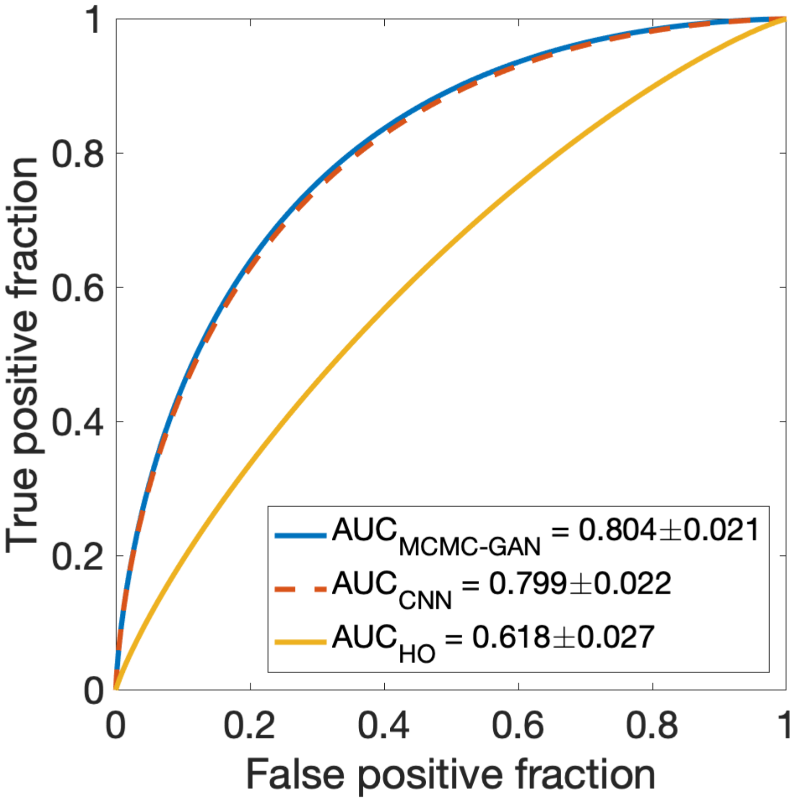


Figure 6.4: The ROC curve corresponding to the MCMC-GAN IO, CNN-IO and the HO. The ROC curve corresponding to the MCMC-GAN IO is in close agreement with the CNN-IO and is higher than the HO.

6.5.2 Clinical brain MR images

The GAN-generated images (bottom row) and ground-truth images (top row) are shown in Fig. 6.5. The ProGAN-generated images have similar visual appearances to the ground-truth ones.

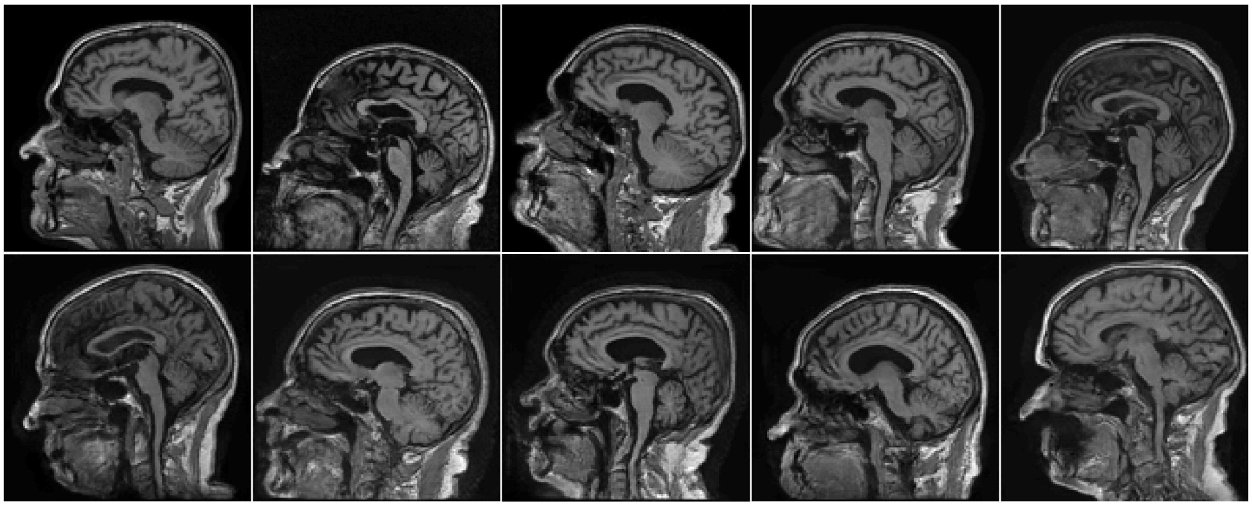


Figure 6.5: Top: Examples of ground-truth MR images. Bottom: Examples of ProGAN-generated MR images.

The ProGAN-generated images have similar visual appearance to the ground-truth images. The Fréchet Inception Distance (FID) [53] score was computed to evaluate the performance of the ProGAN. Lower FID score indicates better quality and diversity of the generated objects. The FID score was 20.9297 that was evaluated on the original 12,000 ground-truth images and 12,000 ProGAN-generated images.

The ROC curves corresponding to the MCMC-GAN IO (blue curve), CNN-IO (red-dashed curve) and the HO (yellow curve) are shown in Fig. 6.6. The curves of the MCMC-GAN IO and CNN-IO are in close agreement, and are higher than the curve of the HO as expected.

The AUC value corresponding to the MCMC-GAN IO, CNN-IO and the HO are 0.861 ± 0.018, 0.857 ± 0.018 and 0.730 ± 0.025, respectively.

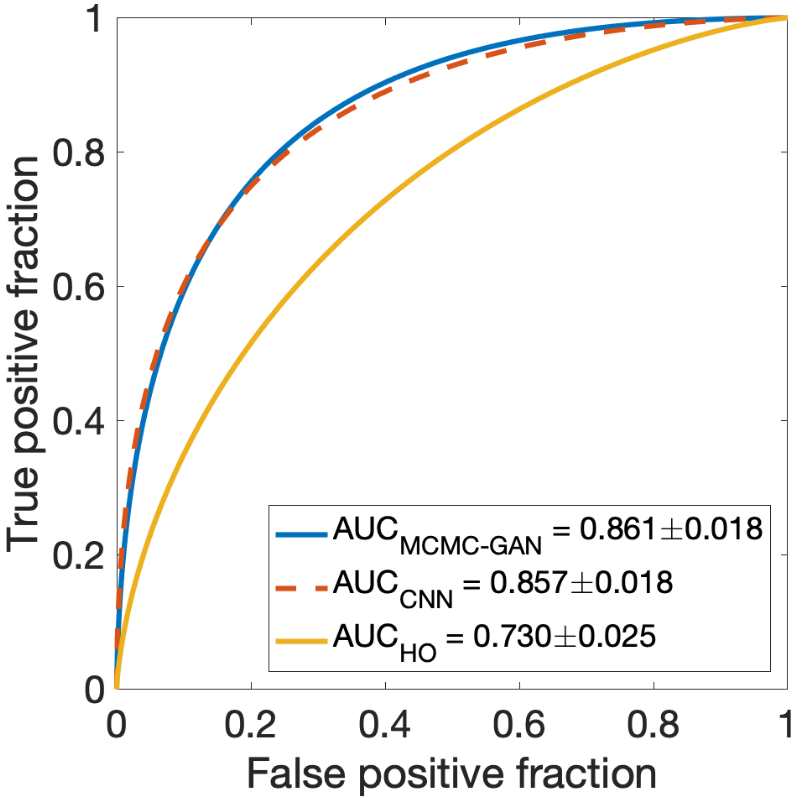


Figure 6.6: The ROC curve corresponding to the MCMC-GAN IO, CNN-IO and the HO. The ROC curve corresponding to the MCMC-GAN IO is in close agreement with the CNN-IO and is higher than the HO.

6.6 Discussion and Conclusion

In this chapter, we proposed a novel sampling-based method that employs MCMC techniques and deep generative models trained by use of GANs to approximate the IO. Although the conventional MCMC methods have been employed to approximate the IO, they have been limited to some relatively simple object models such as lumpy object models, binary texture models and parameterized torso phantoms. Our proposed method extends the domain of applicability of the MCMC techniques and can be implemented with more sophisticated object models. This is because the implementation of GANs is general and not limited to specific images. To demonstrate this, we applied the proposed MCMC-GAN method to the GAN-represented SOMs trained with clinical brain PET images and MR images. The IO performances corresponding to the MCMC-GAN were consistent with those corresponding to the CNN approximated IO. To the best of our knowledge, this is the first time that the MCMC techniques are applied to the SOMs established by use of clinical brain PET and MR data.

It is critical to evaluate the GAN-represented SOMs. In this study, visual assessment of GAN-produced images was conducted and the FID score, a widely used metric to assess GANs in GAN literatures, was reported. However, in medical imaging, it is still important to evaluate GAN-represented SOMs for optimizing imaging systems and data-acquisition designs by use of task-based measures of image quality. For example, a GAN can be potentially assessed by comparing the rank ordering of a set of imaging systems determined by use of a GAN-represented SOM to that produced by use of the ground-truth SOM. Evaluating GANs for optimizing imaging systems by use of task-based measures of image quality represents an important topic for investigation.

There remain additional topics for future investigation. In this study, a random walk Metropolis-Hastings (RWMH) algorithm with a simple Gaussian proposal density function was employed. More advanced MCMC algorithms such as Metropolis adjusted Langevin algorithms (MALA) and Hamiltonian Monte Carlo (HMC) [86] can be readily implemented in our proposed MCMC-GAN framework. This is because the gradient of the generator in GANs with respect to the latent vector can be readily computed on machine learning platforms such as Tensorflow [1]. Finally, it should be noted that our proposed method can be readily applied to approximate the IO for signal detection-localization tasks that were introduced in Chapter 4.

Chapter 7

Summary

In this dissertation, we have developed and investigated machine learning and deep learning methods for assessing task-based measures of image quality (IQ) that quantify the performance of an observer at specific tasks. Supervised learning based methods that employ convolutional neural networks (CNNs) were proposed to approximate the Ideal Observer (IO) for binary signal detection tasks and signal detection-localization tasks. We also proposed supervised learning based methods that employ single layer neural networks (SLNNs) to approximate the Hotelling Observer (HO) without estimating and inverting large covariance matrices. Moreover, a novel deep learning method named progressively-growing Ambient-GANs (ProAmGANs) was developed to establish realistic stochastic object models (SOMs) from noisy imaging measurement data. This method further facilitates computer-simulation for optimizing imaging systems and data-acquisition designs. Finally, a novel sampling based method named MCMC-GAN was developed for approximating the IO. This method can be implemented with sophisticated object models and therefore extends the domain of applicability of MCMC techniques.

A supervised learning-based method that employs CNNs to approximate the IO for binary signal detection tasks was proposed. The considered binary signal detection tasks involved various object models in combination with several measurement noise models. The IO performance that was assessed by the receiver operating characteristic (ROC) curve corresponding to the proposed method was compared to that corresponding to the analytical computation or MCMC method when feasible. Those IO performances were in close agreement. It was also demonstrated that the proposed supervised learning method can still be implemented to a clustered lumpy object model [13] for which the IO computation has not been addressed by the MCMC method. Supervised learning based methods that employ SLNNs were also developed to approximate the HO. These methods directly learn the Hotelling template without estimating and inverting covariance matrices. Accordingly, they can scale well to large images.

Moreover, a supervised learning-based method that employs CNNs to approximate the IO for signal detection-localization tasks was proposed. This method represents a deep-learning-implementation of the IO decision strategy proposed by Khurd and Gindi [62] that optimizes the localization ROC curve (LROC). The considered signal detection-localization tasks involved various object models in combination with several measurement noise models. The observer performance was assessed via the LROC analysis. The LROC curves produced by the proposed supervised-learning method were compared to those produced by the analytical computation or MCMC methods when feasible. Those LROC curves were in close agreement. In addition, it was demonstrated that the proposed supervised learning method can still be implemented to a clustered lumpy object model for which the IO computation has not been addressed by the MCMC method.

An important factor that can significantly limit the performance of an observer is the variation in the objects to-be-imaged. This variation can be described by SOMs. In this dissertation, we developed a deep learning-based method that employed ProAmGANs for establishing SOMs from measured image data. The proposed ProAmGAN strategy incorporates the advanced progressive growing training procedure and therefore enables the AmbientGAN to be applied to realistically sized medical image data. To demonstrate this, stylized numerical studies were conducted in which ProAmGANs were trained on different object ensembles corresponding to common medical imaging modalities. Both visual examinations and quantitative analyses including task-specific validations indicate that the proposed ProAmGANs hold promise to establish realistic SOMs from measured image data.

Moreover, a novel sampling-based method that employs MCMC techniques and deep generative models trained by use of GAN techniques were proposed to approximate the IO. This method applies the MCMC methods to SOMs that are established by use of GAN techniques. Because the implementation of GANs is general and not limited to specific images, our proposed method can be implemented with sophisticated object models and therefore extends the domain of applicability of the MCMC techniques. To demonstrate this, we applied the proposed MCMC-GAN method to the GAN-represented SOMs trained with clinical brain PET images and MR images. To the best of our knowledge, this is the first time that the MCMC techniques are applied to SOMs established by use of clinical brain PET and MR data.

Many topics remain for future investigation. It will be important to quantify the effect of the number of data used in the proposed supervised learning methods for approximating the IO and HO. In addition, to implement the proposed supervised learning methods for approximating the IO in situations where only a limited number of experimental data is available, it will be important to investigate methods to train deep neural networks on limited training data. To achieve this, one may investigate the methods that employ domain adaptation [39, 49] and transfer learning [88]. It will also be important to investigate supervised learning methods for approximating IOs for other more general tasks such as joint signal detection and estimation tasks associated with the estimation ROC (EROC) curve. Moreover, it is critical to evaluate the GAN-represented SOMs for optimizing imaging systems and data-acquisition designs for diagnostic tasks. Finally, it will be important to investigate advanced MCMC algorithms such as Metropolis adjusted Langevin algorithms (MALA) and Hamiltonian Monte Carlo (HMC) [86] for use in our proposed MCMC-GAN framework for approximating the IO.

References

[1] Martín Abadi et al. “Tensorflow: a system for large-scale machine learning.” In: *OSDI*. Vol. 16. 2016, pp. 265–283.

[2] Craig K Abbey and John M Boone. “An Ideal Observer for a model of X-ray imaging in breast parenchymal tissue”. In: *International Workshop on Digital Mammography*. Springer. 2008, pp. 393–400.

[3] M Alnowami et al. “A deep learning model observer for use in alterative forced choice virtual clinical trials”. In: *Medical Imaging 2018: Image Perception, Observer Performance, and Technology Assessment*. Vol. 10577. International Society for Optics and Photonics. 2018, 105770Q.

[4] Felix Ambellan, Alexander Tack, Moritz Ehlke, and Stefan Zachow. “Automated segmentation of knee bone and cartilage combining statistical shape knowledge and convolutional neural networks: Data from the Osteoarthritis Initiative”. In: *Medical Image Analysis* 52 (2019), pp. 109–118.

[5] Mark A Anastasio, Cheng-Ying Chou, Adam M Zysk, and Jovan G Brankov. “Analysis of ideal observer signal detectability in phase-contrast imaging employing linear shift-invariant optical systems”. In: *JOSA A* 27.12 (2010), pp. 2648–2659.

[6] Martin Arjovsky and Léon Bottou. “Towards principled methods for training generative adversarial networks”. In: *International Conference on Learning Representations (ICLR 2017)*. 2017.

[7] Martin Arjovsky, Soumith Chintala, and Léon Bottou. “Wasserstein GAN”. In: *arXiv e-prints*, arXiv:1701.07875 (Jan. 2017), arXiv:1701.07875. arXiv: `1701.07875 [stat.ML]`.

[8] Sanjeev Arora and Yi Zhang. “Do GANs actually learn the distribution? An empirical study”. In: *CoRR* abs/1706.08224 (2017). arXiv: `1706.08224`.

[9] Harrison H Barrett and Kyle J Myers. *Foundations of Image Science*. John Wiley & Sons, 2013.

[10] Harrison H Barrett, Kyle J Myers, Nicholas Devaney, and Christopher Dainty. “Objective assessment of image quality. IV. Application to adaptive optics”. In: *JOSA A* 23.12 (2006), pp. 3080–3105.

[11] Harrison H Barrett, Jie Yao, Jannick P Rolland, and Kyle J Myers. “Model observers for assessment of image quality”. In: *Proceedings of the National Academy of Sciences* 90.21 (1993), pp. 9758–9765.

[12] Sayantan Bhadra, Weimin Zhou, and Mark A Anastasio. “Medical image reconstruction with image-adaptive priors learned by use of generative adversarial networks”. In: *Medical Imaging 2020: Physics of Medical Imaging*. Vol. 11312. International Society for Optics and Photonics. 2020, p. 113120V.

[13] François O Bochud, Craig K Abbey, and Miguel P Eckstein. “Statistical texture synthesis of mammographic images with clustered lumpy backgrounds”. In: *Optics express* 4.1 (1999), pp. 33–43.

[14] Ashish Bora, Ajil Jalal, Eric Price, and Alexandros G Dimakis. “Compressed sensing using generative models”. In: *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org. 2017, pp. 537–546.

[15] Ashish Bora, Eric Price, and Alexandros G Dimakis. “AmbientGAN: Generative models from lossy measurements”. In: *International Conference on Learning Representations (ICLR)*. 2018.

[16] Jovan G Brankov, Yongyi Yang, Liyang Wei, Issam El Naqa, and Miles N Wernick. “Learning a channelized observer for image quality assessment”. In: *IEEE Transactions on Medical Imaging* 28.7 (2009), p. 991.

[17] Andrew Brock, Jeff Donahue, and Karen Simonyan. “Large Scale GAN Training for High Fidelity Natural Image Synthesis”. In: *CoRR* abs/1809.11096 (2018). arXiv: `1809.11096`.

[18] AE Burgess, RF Wagner, RJ Jennings, and Horace B Barlow. “Efficiency of human visual signal discrimination”. In: *Science* 214.4516 (1981), pp. 93–94.

[19] Martin Caon. “Voxel-based computational models of real human anatomy: a review”. In: *Radiation and Environmental Biophysics* 42.4 (2004), pp. 229–235.

[20] C Castella, MP Eckstein, CK Abbey, K Kinkel, FR Verdun, RS Saunders, E Samei, and FO Bochud. “Mass detection on mammograms: influence of signal shape uncertainty on human and model observers”. In: *JOSA A* 26.2 (2009), pp. 425–436.

[21] Jun Cheng. “Brain tumor dataset”. In: ().

[22] Dan CireşAn, Ueli Meier, Jonathan Masci, and Jürgen Schmidhuber. “Multi-column deep neural network for traffic sign classification”. In: *Neural Networks* 32 (2012), pp. 333–338.

[23] Eric Clarkson. “Asymptotic ideal observers and surrogate figures of merit for signal detection with list-mode data”. In: *JOSA A* 29.10 (2012), pp. 2204–2216.

[24] Eric Clarkson. “Estimation receiver operating characteristic curve and ideal observers for combined detection/estimation tasks”. In: *JOSA A* 24.12 (2007), B91–B98.

[25] Eric Clarkson and Harrison H Barrett. “Approximations to Ideal-Observer performance on signal-detection tasks”. In: *Applied Optics* 39.11 (2000), pp. 1783–1793.

[26] Eric Clarkson, Matthew A Kupinski, and Harrison H Barrett. “Transformation of characteristic functionals through imaging systems”. In: *Optics express* 10.13 (2002), pp. 536–539.

[27] Eric Clarkson and Fangfang Shen. “Fisher information and surrogate figures of merit for the task-based assessment of image quality”. In: *JOSA A* 27.10 (2010), pp. 2313–2326.

[28] D Louis Collins, Alex P Zijdenbos, Vasken Kollokian, John G Sled, Noor J Kabani, Colin J Holmes, and Alan C Evans. “Design and construction of a realistic digital brain phantom”. In: *IEEE Transactions on Medical Imaging* 17.3 (1998), pp. 463–468.

[29] TF Cootes, MG Roberts, KO Babalola, and CJ Taylor. “Active Shape and Appearance Models”. In: *Handbook of Biomedical Imaging*. Springer, 2015, pp. 105–122.

[30] Timothy F Cootes, Christopher J Taylor, David H Cooper, and Jim Graham. “Active shape models-their training and application”. In: *Computer vision and image understanding* 61.1 (1995), pp. 38–59.

[31] Corinna Cortes, Xavi Gonzalvo, Vitaly Kuznetsov, Mehryar Mohri, and Scott Yang. “Adanet: Adaptive structural learning of artificial neural networks”. In: *arXiv preprint arXiv:1607.01097* (2016).

[32] Emily L. Denton, Soumith Chintala, Arthur Szlam, and Robert Fergus. “Deep Generative Image Models using a Laplacian Pyramid of Adversarial Networks”. In: *CoRR* abs/1506.05751 (2015). arXiv: `1506.05751`.

[33] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. “Learning a deep convolutional network for image super-resolution”. In: *European Conference on Computer Vision*. Springer. 2014, pp. 184–199.

[34] Miguel P Eckstein and Craig K Abbey. “Model observers for signal-known-statistically tasks (SKS)”. In: *Medical Imaging 2001: Image Perception and Performance*. Vol. 4324. International Society for Optics and Photonics. 2001, pp. 91–103.

[35] Anders Eklund. “Feeding the zombies: Synthesizing brain volumes using a 3D progressive growing GAN”. In: *arXiv preprint arXiv:1912.05357* (2019).

[36] Fatma Elzahraa A Elshahaby, Michael Ghaly, Abhinav K Jha, and Eric C Frey. “Factors affecting the normality of channel outputs of channelized model observers: an investigation using realistic myocardial perfusion SPECT images”. In: *Journal of Medical Imaging* 3.1 (2016), p. 015503.

[37] Vittorio Ferrari, Frederic Jurie, and Cordelia Schmid. “From images to shape models for object detection”. In: *International Journal of Computer Vision* 87.3 (2010), pp. 284–303.

[38] Brandon D Gallas and Harrison H Barrett. “Validating the use of channels to estimate the ideal linear observer”. In: *JOSA A* 20.9 (2003), pp. 1725–1738.

[39] Yaroslav Ganin and Victor Lempitsky. "Unsupervised domain adaptation by backpropagation". In: *arXiv preprint arXiv:1409.7495* (2014).

[40] Christophe Garcia and Manolis Delakis. "Convolutional face finder: A neural architecture for fast and robust face detection". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26.11 (2004), pp. 1408–1423.

[41] HC Gifford. "Efficient visual-search model observers for PET". In: *The British journal of radiology* 87.1039 (2014), p. 20140017.

[42] Howard C Gifford, Michael A King, P Hendrik Pretorius, and R Glenn Wells. "A comparison of human and model observers in multislice LROC studies". In: *IEEE transactions on medical imaging* 24.2 (2005), pp. 160–169.

[43] Howard C Gifford, Zhihua Liang, and Mini Das. "Visual-search observers for assessing tomographic x-ray image quality". In: *Medical physics* 43.3 (2016), pp. 1563–1575.

[44] Howard C Gifford, RG Wells, and Michael A King. "A comparison of human observer LROC and numerical observer ROC for tumor detection in SPECT images". In: *IEEE Transactions on Nuclear Science* 46.4 (1999), pp. 1032–1037.

[45] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. The MIT Press, 2016.

[46] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. "Generative adversarial nets". In: *Advances in Neural Information Processing Systems*. 2014, pp. 2672–2680.

[47] Nelly Gordillo, Eduard Montseny, and Pilar Sobrevilla. "State of the art survey on MRI brain tumor segmentation". In: *Magnetic resonance imaging* 31.8 (2013), pp. 1426–1438.

[48] Ishaan Gulrajani, Faruk Ahmed, Martín Arjovsky, Vincent Dumoulin, and Aaron C. Courville. "Improved Training of Wasserstein GANs". In: *CoRR* abs/1704.00028 (2017). arXiv: `1704.00028`.

[49] Shenghua He, Weimin Zhou, Hua Li, and Mark A Anastasio. "Learning numerical observers using unsupervised domain adaptation". In: *Medical Imaging 2020: Image Perception, Observer Performance, and Technology Assessment*. Vol. 11316. International Society for Optics and Photonics. 2020, 113160W.

[50] Xin He, Brian S Caffo, and Eric C Frey. "Toward realistic and practical Ideal Observer (IO) estimation for the optimization of medical imaging systems". In: *IEEE Transactions on Medical Imaging* 27.10 (2008), pp. 1535–1543.

[51] Tobias Heimann and Hans-Peter Meinzer. "Statistical shape models for 3D medical image segmentation: a review". In: *Medical Image Analysis* 13.4 (2009), pp. 543–563.

[52] John J Heine, Stanley R Deans, and Laurence P Clarke. "Multiresolution probability analysis of random fields". In: *JOSA A* 16.1 (1999), pp. 6–16.

[53] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. “Gans trained by a two time-scale update rule converge to a local nash equilibrium”. In: *Advances in Neural Information Processing Systems.* 2017, pp. 6626–6637.

[54] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. “Multilayer feedforward networks are universal approximators”. In: *Neural Networks* 2.5 (1989), pp. 359–366.

[55] Shady Abu Hussein, Tom Tirer, and Raja Giryes. “Image-adaptive gan based reconstruction”. In: *arXiv preprint arXiv:1906.05284* (2019).

[56] Abhinav K Jha, Eric Clarkson, and Matthew A Kupinski. “An ideal-observer framework to investigate signal detectability in diffuse optical imaging”. In: *Biomedical optics express* 4.10 (2013), pp. 2107–2123.

[57] Philip F. Judy. *LROC Software.* 2010.

[58] Avinash C Kak, Malcolm Slaney, and Ge Wang. “Principles of computerized tomographic imaging”. In: *Medical Physics* 29.1 (2002), pp. 107–107.

[59] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. “Progressive Growing of GANs for improved quality, stability, and variation”. In: *arXiv preprint arXiv:1710.10196* (2017).

[60] Tero Karras, Samuli Laine, and Timo Aila. “A style-based generator architecture for generative adversarial networks”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 2019, pp. 4401–4410.

[61] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. “Analyzing and improving the image quality of stylegan”. In: *arXiv preprint arXiv:1912.04958* (2019).

[62] Parmeshwar Khurd and Gene Gindi. “Decision strategies that maximize the area under the LROC curve”. In: *IEEE Transactions on Medical Imaging* 24.12 (2005), pp. 1626–1636.

[63] Diederik P Kingma and Jimmy Ba. “Adam: A method for stochastic optimization”. In: *arXiv preprint arXiv:1412.6980* (2014).

[64] Matthew A Kupinski and Harrison H Barrett. *Small-animal SPECT imaging.* Vol. 233. Springer, 2005.

[65] Matthew A Kupinski, Eric Clarkson, and Jacob Y Hesterman. “Bias in Hotelling observer performance computed from finite data”. In: *Medical Imaging 2007: Image Perception, Observer Performance, and Technology Assessment.* Vol. 6515. International Society for Optics and Photonics. 2007, 65150S.

[66] Matthew A Kupinski, Eric Clarkson, John W Hoppin, Liying Chen, and Harrison H Barrett. “Experimental determination of object statistics from noisy images”. In: *JOSA A* 20.3 (2003), pp. 421–429.

[67] Matthew A Kupinski, Darrin C Edwards, Maryellen L Giger, and Charles E Metz. "Ideal Observer approximation using Bayesian classification neural networks". In: *IEEE Transactions on Medical Imaging* 20.9 (2001), pp. 886–899.

[68] Matthew A Kupinski, John W Hoppin, Eric Clarkson, and Harrison H Barrett. "Ideal-Observer computation in medical imaging with use of Markov-Chain Monte Carlo techniques". In: *JOSA A* 20.3 (2003), pp. 430–438.

[69] Samuel J LaRoque, Emil Y Sidky, Darrin C Edwards, and Xiaochuan Pan. "Evaluation of the channelized Hotelling observer for signal detection in 2D tomographic imaging". In: *Medical Imaging 2007: Image Perception, Observer Performance, and Technology Assessment*. Vol. 6515. International Society for Optics and Photonics. 2007, p. 651514.

[70] Steve Lawrence, C Lee Giles, Ah Chung Tsoi, and Andrew D Back. "Face recognition: A convolutional neural-network approach". In: *IEEE Transactions on Neural Networks* 8.1 (1997), pp. 98–113.

[71] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. "Deep learning". In: *Nature* 521.7553 (2015), p. 436.

[72] Christina M Li, W Paul Segars, Georgia D Tourassi, John M Boone, and James T Dobbins III. "Methodology for generating a 3D computerized breast phantom from empirical data". In: *Medical Physics* 36.7 (2009), pp. 3122–3131.

[73] Steven Cheng-Xian Li, Bo Jiang, and Benjamin M. Marlin. "MisGAN: Learning from Incomplete Data with Generative Adversarial Networks". In: *CoRR* abs/1902.09599 (2019). arXiv: `1902.09599`.

[74] Xin Li, Abhinav K Jha, Michael Ghaly, Fatma EA Elshahaby, Jonathan M Links, and Eric C Frey. "Use of sub-ensembles and multi-template observers to evaluate detection task performance for data that are not multivariate normal". In: *IEEE transactions on medical imaging* 36.4 (2016), pp. 917–929.

[75] Francesc Massanes and Jovan G Brankov. "Evaluation of CNN as anthropomorphic model observer". In: *Medical Imaging 2017: Image Perception, Observer Performance, and Technology Assessment*. Vol. 10136. International Society for Optics and Photonics. 2017, 101360Q.

[76] CE Metz. "Rockit User's guide". In: *Chicago, Department of Radiology, University of Chicago* (1998).

[77] Charles E Metz and Xiaochuan Pan. "'Proper' binormal ROC curves: theory and maximum-likelihood estimation". In: *Journal of Mathematical Psychology* 43.1 (1999), pp. 1–33.

[78] Susanne G Mueller et al. "The Alzheimer's disease neuroimaging initiative". In: *Neuroimaging Clinics* 15.4 (2005), pp. 869–877.

[79] KJ Myers, JP Rolland, Harrison H Barrett, and RF Wagner. "Aperture optimization for emission imaging: effect of a spatially varying background". In: *JOSA A* 7.7 (1990), pp. 1279–1293.

[80] Kyle J Myers, Robert F Wagner, and Kenneth M Hanson. "Rayleigh task performance in tomographic reconstructions: Comparison of human and machine performance". In: *Medical Imaging 1993: Image Processing*. Vol. 1898. International Society for Optics and Photonics. 1993, pp. 628–637.

[81] Subok Park, Harrison H Barrett, Eric Clarkson, Matthew A Kupinski, and Kyle J Myers. "Channelized-Ideal Observer using Laguerre-Gauss channels in detection tasks involving non-Gaussian distributed lumpy backgrounds and a Gaussian signal". In: *JOSA A* 24.12 (2007), B136–B150.

[82] Subok Park and Eric Clarkson. "Efficient estimation of Ideal-Observer performance in classification tasks involving high-dimensional complex backgrounds". In: *JOSA A* 26.11 (2009), B59–B71.

[83] Subok Park, Eric Clarkson, Matthew A Kupinski, and Harrison H Barrett. "Efficiency of the human observer detecting random signals in random backgrounds". In: *JOSA A* 22.1 (2005), pp. 3–16.

[84] Subok Park, Brandon D Gallas, Aldo Badano, Nicholas A Petrick, and Kyle J Myers. "Efficiency of the human observer for detecting a Gaussian signal at a known location in non-Gaussian distributed lumpy backgrounds". In: *JOSA A* 24.4 (2007), pp. 911–921.

[85] Subok Park, Matthew A Kupinski, Eric Clarkson, and Harrison H Barrett. "Ideal-Observer performance under signal and background uncertainty". In: *Biennial International Conference on Information Processing in Medical Imaging*. Springer. 2003, pp. 342–353.

[86] Marcelo Pereyra, Philip Schniter, Emilie Chouzenoux, Jean-Christophe Pesquet, Jean-Yves Tourneret, Alfred O Hero, and Steve McLaughlin. "A survey of stochastic simulation and optimization methods in signal processing". In: *IEEE Journal of Selected Topics in Signal Processing* 10.2 (2015), pp. 224–241.

[87] Lorenzo L Pesce and Charles E Metz. "Reliable and computationally efficient maximum-likelihood estimation of "proper" binormal ROC curves". In: *Academic Radiology* 14.7 (2007), pp. 814–829.

[88] Junfei Qiu, Qihui Wu, Guoru Ding, Yuhua Xu, and Shuo Feng. "A survey of machine learning for big data processing". In: *EURASIP Journal on Advances in Signal Processing* 2016.1 (2016), p. 67.

[89] Alec Radford, Luke Metz, and Soumith Chintala. "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks". In: *arXiv e-prints*, arXiv:1511.06434 (Nov. 2015), arXiv:1511.06434. arXiv: `1511.06434 [cs.LG]`.

[90] Waseem Rawat and Zenghui Wang. "Deep convolutional neural networks for image classification: A comprehensive review". In: *Neural Computation* 29.9 (2017), pp. 2352–2449.

[91] J. P. Rolland and H. H. Barrett. “Effect of random background inhomogeneity on observer detection performance”. In: *J. Opt. Soc. Am. A* 9.5 (May 1992), pp. 649–658. DOI: `10.1364/JOSAA.9.000649`.

[92] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. “Improved Techniques for Training GANs”. In: *Proceedings of the 30th International Conference on Neural Information Processing Systems*. NIPS’16. Barcelona, Spain: Curran Associates Inc., 2016, pp. 2234–2242.

[93] Dominik Scherer, Andreas Müller, and Sven Behnke. “Evaluation of pooling operations in convolutional architectures for object recognition”. In: *Artificial Neural Networks–ICANN 2010*. Springer, 2010, pp. 92–101.

[94] W Paul Segars and Benjamin MW Tsui. “Study of the efficacy of respiratory gating in myocardial SPECT using the new 4-D NCAT phantom”. In: *IEEE Transactions on Nuclear Science* 49.3 (2002), pp. 675–679.

[95] William Paul Segars, M Mahesh, Thomas J Beck, Eric C Frey, and Benjamin MW Tsui. “Realistic CT simulation using the 4D XCAT phantom”. In: *Medical Physics* 35.8 (2008), pp. 3800–3808.

[96] Fangfang Shen and Eric Clarkson. “Using Fisher information to approximate Ideal-Observer performance on detection tasks for lumpy-background images”. In: *JOSA A* 23.10 (2006), pp. 2406–2414.

[97] Kai-kai Shen, Jurgen Fripp, Fabrice Mériaudeau, Gaël Chételat, Olivier Salvado, Pierrick Bourgeat, Alzheimer’s Disease Neuroimaging Initiative, et al. “Detecting global and local hippocampal shape changes in Alzheimer’s disease using statistical shape models”. In: *Neuroimage* 59.3 (2012), pp. 2155–2166.

[98] Ashish Shrivastava, Tomas Pfister, Oncel Tuzel, Josh Susskind, Wenda Wang, and Russell Webb. “Learning from Simulated and Unsupervised Images through Adversarial Training”. In: *CoRR* abs/1612.07828 (2016). arXiv: `1612.07828`.

[99] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. “Striving for simplicity: The all convolutional net”. In: *arXiv preprint arXiv:1412.6806* (2014).

[100] Stuart J Starr, Charles E Metz, Lee B Lusted, and David J Goodenough. “Visual detection and localization of radiographic images”. In: *Radiology* 116.3 (1975), pp. 533–538.

[101] Richard G Swensson. “Unified measurement of observer performance in detecting and localizing target objects on images”. In: *Medical physics* 23.10 (1996), pp. 1709–1725.

[102] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. “Rethinking the inception architecture for computer vision”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 2818–2826.

[103] Sho Tomoshige, Elco Oost, Akinobu Shimizu, Hidefumi Watanabe, and Shigeru Nawano. “A conditional statistical shape model with integrated error estimation of the conditions; application to liver segmentation in non-contrast CT images”. In: *Medical Image Analysis* 18.1 (2014), pp. 130–143.

[104] Robert F Wagner and David G Brown. “Unified SNR analysis of medical imaging systems”. In: *Physics in Medicine & Biology* 30.6 (1985), p. 489.

[105] Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, Mohammadhadi Bagheri, and Ronald M Summers. “Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 2097–2106.

[106] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. “Image quality assessment: from error visibility to structural similarity”. In: *IEEE Transactions on Image Processing* 13.4 (2004), pp. 600–612.

[107] Miles N Wernick, Yongyi Yang, Jovan G Brankov, Grigori Yourganov, and Stephen C Strother. “Machine learning in medical imaging”. In: *IEEE Signal Processing Magazine* 27.4 (2010), pp. 25–38.

[108] Adam Wunderlich, Bart Goossens, and Craig K Abbey. “Optimal joint detection and estimation that maximizes ROC-type curves”. In: *IEEE transactions on medical imaging* 35.9 (2016), pp. 2164–2173.

[109] X George Xu. “An exponential growth of computational phantom research in radiation protection, imaging, and radiotherapy: a review of the fifty-year history”. In: *Physics in medicine and biology* 59.18 (2014), R233.

[110] Ke Yan, Xiaosong Wang, Le Lu, and Ronald M Summers. “DeepLesion: automated mining of large-scale lesion annotations and universal lesion detection with deep learning”. In: *Journal of Medical Imaging* 5.3 (2018), p. 036501.

[111] Ian T Young. “Proof without prejudice: use of the Kolmogorov-Smirnov test for the analysis of histograms from flow systems and other sources.” In: *Journal of Histochemistry & Cytochemistry* 25.7 (1977), pp. 935–941.

[112] Maria Zankl and K Eckerman. “The GSF voxel computational phantom family”. In: *Handbook of Anatomical Models for Radiation Dosimetry* (2010), pp. 65–85.

[113] Jure Zbontar et al. “fastMRI: An open dataset and benchmarks for accelerated MRI”. In: *arXiv preprint arXiv:1811.08839* (2018).

[114] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. “Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising”. In: *IEEE Transactions on Image Processing* 26.7 (2017), pp. 3142–3155.

[115] Yani Zhang, Binh T Pham, and Miguel P Eckstein. “Automated optimization of JPEG 2000 encoder options based on model observer performance for detecting variable signals in X-ray coronary angiograms”. In: *IEEE Transactions on Medical Imaging* 23.4 (2004), pp. 459–474.

[116] Lili Zhou and Gene Gindi. “Collimator optimization in SPECT based on a joint detection and localization task”. In: *Physics in Medicine & Biology* 54.14 (2009), p. 4423.

[117] Weimin Zhou and Mark A Anastasio. “Learning the ideal observer for joint detection and localization tasks by use of convolutional neural networks”. In: *Medical Imaging 2019: Image Perception, Observer Performance, and Technology Assessment*. Vol. 10952. International Society for Optics and Photonics. 2019, p. 1095209.

[118] Weimin Zhou and Mark A Anastasio. “Learning the Ideal Observer for SKE detection tasks by use of convolutional neural networks”. In: *Medical Imaging 2018: Image Perception, Observer Performance, and Technology Assessment*. Vol. 10577. International Society for Optics and Photonics. 2018, p. 1057719.

[119] Weimin Zhou and Mark A Anastasio. “Markov-Chain Monte Carlo approximation of the Ideal Observer using generative adversarial networks”. In: *Medical Imaging 2020: Image Perception, Observer Performance, and Technology Assessment*. Vol. 11316. International Society for Optics and Photonics. 2020, p. 113160D.

[120] Weimin Zhou, Sayantan Bhadra, Frank J Brooks, Hua Li, and Mark A Anastasio. “Learning stochastic object models from medical imaging measurements using Progressively-Growing AmbientGANs.” In: *arXiv preprint arXiv:2006.00033* (2020).

[121] Weimin Zhou, Sayantan Bhadra, Frank J Brooks, Hua Li, and Mark A Anastasio. “Progressively-Growing AmbientGANs for learning stochastic object models from imaging measurements”. In: *Medical Imaging 2020: Image Perception, Observer Performance, and Technology Assessment*. Vol. 11316. International Society for Optics and Photonics. 2020, 113160Q.

[122] Weimin Zhou, Sayantan Bhadra, Frank Brooks, and Mark A Anastasio. “Learning stochastic object model from noisy imaging measurements using AmbientGANs”. In: *Medical Imaging 2019: Image Perception, Observer Performance, and Technology Assessment*. Vol. 10952. International Society for Optics and Photonics. 2019, p. 109520M.

[123] Weimin Zhou, Hua Li, and Mark A Anastasio. “Approximating the Ideal Observer and Hotelling Observer for binary signal detection tasks by use of supervised learning methods”. In: *IEEE Transactions on Medical Imaging* 38.10 (2019), pp. 2456–2468.

[124] Weimin Zhou, Hua Li, and Mark A Anastasio. “Approximating the Ideal Observer for joint signal detection and localization tasks by use of supervised learning methods”. In: *IEEE Transactions on Medical Imaging* (2020), pp. 1–1.

[125] Weimin Zhou, Hua Li, and Mark A Anastasio. "Learning the Hotelling observer for SKE detection tasks by use of supervised learning methods". In: *Medical Imaging 2019: Image Perception, Observer Performance, and Technology Assessment*. Vol. 10952. International Society for Optics and Photonics. 2019, p. 1095208.

[126] X George Zu. "The VIP-Man Model-A Digital Human Testbed for Radiation Siimulations". In: *SAE Transactions* (2005), pp. 779–787.

Appendix A

Gradient of cross-entropy

The cross-entropy can be written as:

$$\begin{aligned}
&\langle -\log[\Pr(H_y|\mathbf{g},\boldsymbol{\Theta})]\rangle_{(\mathbf{g},y)} \\
&= -\int d\mathbf{g}\ \left[\sum_{y=0}^{J} p(\mathbf{g},H_y)\log\frac{\exp\left[z_y(\mathbf{g};\boldsymbol{\Theta})\right]}{\sum_{j'=0}^{J}\exp\left[z_{j'}(\mathbf{g};\boldsymbol{\Theta})\right]}\right] \\
&= -\int d\mathbf{g}\ \Big\{\sum_{y=0}^{J} p(\mathbf{g},H_y) z_y(\mathbf{g};\boldsymbol{\Theta}) \\
&\qquad - \sum_{y=0}^{J} p(\mathbf{g},H_y)\log\big(\sum_{j'=0}^{J}\exp\left[z_{j'}(\mathbf{g};\boldsymbol{\Theta})\right]\big)\Big\}.
\end{aligned} \tag{A.1}$$

Here, the cross-entropy $\langle -\log[\Pr(H_y|\mathbf{g},\boldsymbol{\Theta})]\rangle_{(\mathbf{g},y)}$ is considered as a functional of the $z_j(\mathbf{g};\boldsymbol{\Theta})$, viewed as functions of $\mathbf{g}$. The derivative of $\langle -\log[\Pr(H_y|\mathbf{g},\boldsymbol{\Theta})]\rangle_{(\mathbf{g},y)}$ with respect to $z_j(\mathbf{g};\boldsymbol{\Theta})$, which is a functional derivative known as a Fréchet derivative, can subsequently be computed as:

$$\begin{aligned}
&\frac{\partial \langle -\log[\Pr(H_y|\mathbf{g}, \boldsymbol{\Theta})]\rangle_{(\mathbf{g},y)}}{\partial z_j(\mathbf{g}; \boldsymbol{\Theta})} \\
&= -p(\mathbf{g}, H_j) + \sum_{y=0}^{J} p(\mathbf{g}, H_y) \frac{\exp\left[z_j(\mathbf{g}; \boldsymbol{\Theta})\right]}{\sum_{j'=0}^{J} \exp\left[z_{j'}(\mathbf{g}; \boldsymbol{\Theta})\right]} \\
&= -p(\mathbf{g}) \left[p(H_j|\mathbf{g}) - \frac{\exp\left[z_j(\mathbf{g}; \boldsymbol{\Theta})\right]}{\sum_{j'=0}^{J} \exp\left[z_{j'}(\mathbf{g}; \boldsymbol{\Theta})\right]} \right].
\end{aligned} \tag{A.2}$$

The last step in Eq. (A.2) is derived because $p(\mathbf{g}, H_j) = p(\mathbf{g})p(H_j|\mathbf{g})$ and $\sum_{y=0}^{J} p(\mathbf{g}, H_y) = p(\mathbf{g})$.