Low-Dose CT Image Denoising via Deep Learning

MSc Thesis in Bio- and Medical Physics (60 ECTS)

a report by

Ina Michelle Marie Fridlund

Contact: xjg423@alumni.ku.dk, ina.michelle.marie.fridlund@regionh.dk

Supervisors: Lise Arleth[†], Flemming Littrup Andersen[*], Claes Nøhr Ladefoged[*], Sofie Lindskov Hansen[*]
Censor: Søren Baarsgaard Hansen[**]

Date: June 2020

Abstract

The presented thesis was written as a partial requirement for acquiring a Masters degree in Bio- and Medical Physics from the University of Copenhagen, Denmark. The amount of work equated to 60 ECTS points and had been completed in the period from September 2019 to May 2020. The project was conducted in collaboration with the Department of Clinical Physiology, Nuclear Medicine & PET at Rigshospitalet, Copenhagen University Hospital and based at the Clinically Applied Artificial Intelligence Group.

The project focused on investigating the feasibility of a deep learning-based approach to reducing noise inherent to low-dose CT images for quality enhancement. The generative adversarial network (GAN) architectures in particular, namely those recently proposed by *Yi et al.*, 2017 (*J Digit Imaging*) and *Yang et al.*, 2018 (*IEEE Trans. Med. Imaging.*), were scrutinised. The latter was implemented for training a model on a local clinical data of spatially-aligned low- and high- dose CT image pairs.

The resulting denoising method yielded an increase in peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) values of the output images. Moreover, the standard deviations across similar uniform regions were significantly lower for generated images as compared to their low dose counterparts, approaching that of a diagnostic dose. The resulting denoised images received a higher rating for visually-perceived quality by the on-site radiologist: 1.36 ± 0.12 and 1.52 ± 0.12 ($p = 0.08$) out of 3.0 (target) for low-dose and network output images respectively.

Niels Bohr Institute, University of Copenhagen[†]
Department of Clinical Physiology, Nuclear Medicine & PET and Cluster for Molecular Imaging, Rigshospitalet[*]
Department of Nuclear Medicine & PET, Aarhus University Hospital[**]

Contents

* * *

The second part of the presented MSc thesis project had been carried out during the outbreak of the COVID-19 disease. As a direct result of the global pandemic and, indirectly, through the appropriate response of the Danish government, the continuous data acquisition process that provided the primary data for the project had been halted indefinitely prior to the analysis conclusion. While the integrity of the project itself had not been jeopardised, the state of affairs and general lack of amenities during national lockdown affected the methodology and potential results due to limited time and resources. The final decisions were made with the hope of getting the best possible outcome in given circumstances.

1 Introduction

Computed Tomography (CT) is an essential and widely used imaging modality in a clinical environment as well as other fields of application. Where diagnostic radiology is concerned, CT is particularly useful for its high spatial resolution and the three-dimensional rendering that allows to recover the structural depth, lost in a two-dimensional projection image. [1] Unfortunately, there exists an obvious drawback in the form of ionising radiation exposure to patients. [2] CT accounts for approximately 9% of all radiological examinations but is responsible for 47% of medical radiation dose. [3] The associated risks (effective dose) can be limited through either lowering the energy (tube potential in kV) or the flux of the radiation field (tube current in mAs). However, low voltage will result in poor transmission as the photons will not have sufficient energy and lowering the X-ray tube current leads to significantly degraded image quality due to an increased level of quantum noise. This creates a niche for developing a method that would perform accurate denoising without impeding the diagnostic quality. A number of approaches have been developed in recent years to tackle the aforementioned problem, including those in the sinogram domain and complimentary to image reconstruction. [4], [5] The issue with the former is susceptibility to edge blurring and sinogram data of commercial scanners not being readily available to users, while the latter tends to be vendor-specific and computationally cumbersome. Alternatively, post-processing techniques are able to mitigate these difficulties through operating on an image directly without having to rely on raw data. [6], [7]

The image reconstruction techniques in CT are broadly categorised into Fourier-based (Filtered Backprojection or FBP) and iterative (a more detailed account of both can be found in Sections 2.3.2 and 2.3.3). The former involves convolving discretely sampled projection data contributions with appropriate filters prior to transformation into the image domain. Apart from the x-ray beam and the electronics, the choice of filter employed in FBP can also affect the level of noise and even the uniformity of its propagation in the final image. [8] Nevertheless, the FBP algorithm is the one most commonly used in CT scanners. [1] It utilises directly measured attenuations of X-ray transmission lines, while iterative methods evaluate their values numerically and are better suited for imaging modalities with "poorer" photon statistics like SPECT. [9] It is also important to note that even though iterative reconstruction has the potential to serve as an excellent denoising method in itself [10], correctly estimating noise during iterations is not a straight-forward task. [11] Furthermore, iterative reconstruction is associated with producing coarser noise granularity [12] and a visual change in texture that does not appeal to trained radiologists, potentially jeopardising diagnostic confidence. [4], [13] Therefore, one of the goals of this project was to produce model outputs visually similar to those obtained from FBP at higher dose levels.

The growing prominence of Deep Learning (DL) research [14] and its numerous successful applications in the computer vision domain [15] have also lead to adopting deep neural networks as an alternative approach to tackling modern day challenges in the field of medical imaging. [16], [17], [18]

As far as CT images are concerned, the oversmoothing and residual error issues, commonly observed in image processing tasks, are particularly hard to address due to non-uniform noise distribution in reconstructed CT. The advantage of learning-based methods is their immunity to this specific problem as they depend primarily on training samples. [19] In the recent years, a number of Deep Learning approaches to LDCT denoising have been proposed (for example, [21], [22], [23]). Training a noise-reducing neural network typically requires spatially-aligned dignostic (also referred to as routine or normal - NDCT) and low dose CT image pairs (LDCT), which are not always available due to additional radiation exposure to patients and general susceptibility to motion artefacts. *Wolterink et al.* [20] suggested that disregarding the per-voxel loss and only utilising the adversarial feedback from a Generative Adversarial Network or GAN (see Section 2.7 for a more detailed account of GAN arhitecture) could be sufficient for training a robust model. The GAN denoising algorithm is primarily concerned with learning the low-dose image noise distribution with the aim of subtracting it in a way that would mimic the dignostic dose, which acts as the training target (also called the *ground truth* image). Thus, this particular method is also immune to metal artefacts as these features are inherently different to the noise structure. *Yi et al.* [19] and *Yang et al.* [6] tackled the sharpness impairment caused by the mean squared error loss through incorporating auxiliary "sharpness" and "perceptual" losses respectively to a GAN with Wasserstein distance.

This thesis project investigated the feasibility of a post-processing deep learning LDCT image denoising approach. The WGAN-VGG network [6] was trained on a local clinical dataset comprising spatially-aligned LDCT and NDCT image pairs. The model was calibrated and adjusted for best performance on the unique clinical dataset. The denoising ability of the resulting model was analysed quantitatively and qualitatively and results compared to those achieved previously on a simulated dataset. Additionally, network performance was evaluated against that of a different architecture (see [19]) and validated on a simulated dataset, obtained by manually applying a range of noise levels to the high dose images from the test set. The viability of potential clinical implementation was explored by conducting a subjective blinded scoring test, where the overall image quality across low/high dose and resulting denoised network output counterparts were rated by an on-site radiologist. Finally, considerations of limiting factors and suggestions for further model improvements and deep learning noise reduction approaches were discussed.

1.1 Aim

The main aim of the project was to implement the WGAN-VGG architecture, proposed by *Yang et al.* [6] and adjust the network for best results on the local dataset with the ultimate goal of upscaling the quality of LDCT to ease the trade-off between image quality and radiation dose.

2 Theoretical Background

This section will cover the physical, mathematical and technological prerequisites that are essential for an in depth understanding of the underlying theory of the presented thesis project and form the premise for the relevant methodology.

2.1 X-Ray Physics

Diagnostic radiology with X-rays manifests in a range of modalities from CT scans and fluoroscopy to mammography, dental and conventional X-ray examinations. The fundamental principle behind all these imaging techniques is based on registering the distribution of X-ray photons transmitted through an object. The difference in transmitted intensities possesses the information about the attenuating properties of all tissue types traversed before reaching the detector.

2.1.1 X-ray Photon Interactions in Matter

As an X-ray photon passes through matter it interacts with it either by complete absorption, elastic/inelastic scattering or pair production. The mode of interaction depends heavily on the incident photon energy and the absorber material. Figure 1 describes schematically the three dominant modes and their prevalence for different materials and energy ranges. Figure 2 shows the X-ray absorption probabilities for a range of selected elements.

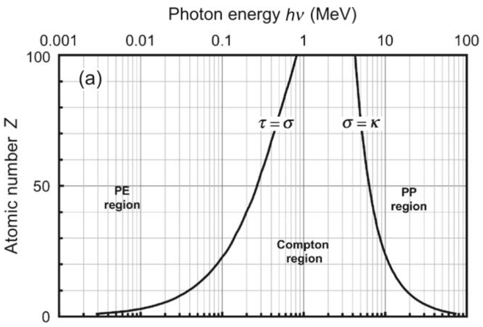


Figure 1: Light-matter interaction types (left to right): photoelectric absorption (PE), Compton scattering and pair production (PP). The two lines correspond to energies and material atomic numbers for which neighbouring interactions are equally likely. Figure from *Podgorsak* (2006) [24].

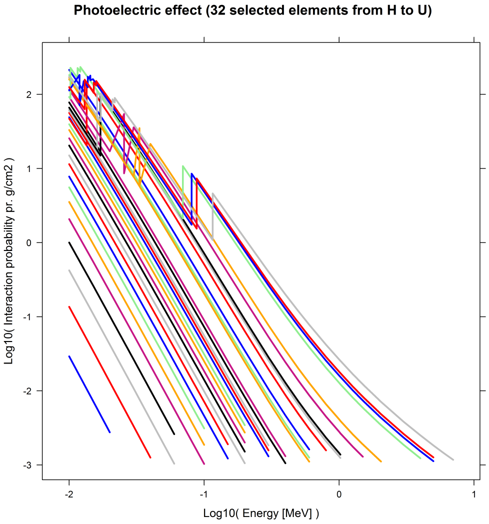


Figure 2: PE interaction probabilities per unit density for a range of photon energies for 32 selected elements (logarithmic scale). The abrupt increase in X-ray absorption occurs when the photon energies are close to absorber atom K-shell binding energy (referred to as *K-edge*). Image adapted from *Dance et al.* (2014) [25].

The predominant interaction mode for lower energy photons is photoelectric absorption. The incident X-ray energy E_γ is transferred into releasing a photoelectron from its bound shell. A characteristic X-ray is emitted as the atom "relaxes" back to its stable state (see Figure 3). The probability of photoelectric absorption is enhanced for materials of high atomic number $\mathbb{Z}$ and diminishes with higher photon energies. This is why high $\mathbb{Z}$ materials such as lead are typically used for gamma-ray shielding. Moreover, distinct attenuation probabilities and K-edges (see Figure 2) for each given $\mathbb{Z}$ facilitate good radiological contrast (see Section 2.4) for differentiating between materials (like bone and soft tissue) at lower photon energies. [24], [26]

The main interaction mechanism for X-rays of intermediate energy range is Compton scattering. During this process, the incident photon of energy $E = h\upsilon$ is deflected off an electron in a material by some angle θ

with respect to its original direction, depositing some of its energy (schematically depicted in Figure 4). The subsequent increase in photon wavelength is given by the Compton equation via elastic momentum conservation:

$$\Delta\lambda = \lambda_s - \lambda_i = \frac{h}{m_e c}(1 - \cos\theta), \tag{1}$$

where λ_s and λ_i are the wavelengths of the scattered and incident photons respectively, h is the Planck's constant, m_e= 511 keV/c^2 is the electron rest-mass and θ is the scattering angle. Since the probability of Compton scattering increases with the number of electrons available, it, therefore, increases linearly with $\mathbb{Z}$, but otherwise depends weakly on the material type and, thus, is of less importance for the purposes of diagnostic imaging. [27]

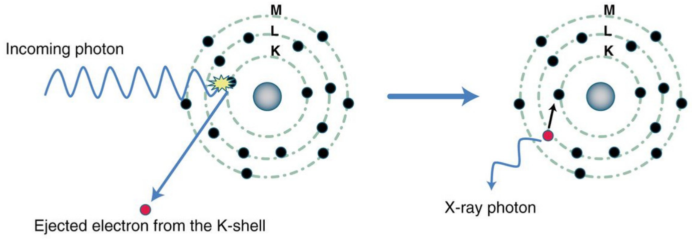


Figure 3: Schematic of photoelectric absorption. If the incident X-ray energy is higher than the binding energy of an electron in a given shell of the absorber atom, a vacancy is created. Characteristic X-rays are emitted during subsequent state transitions as the vacancy is filled by one of the higher shell electrons. Image from of *Danad et al.* [28].

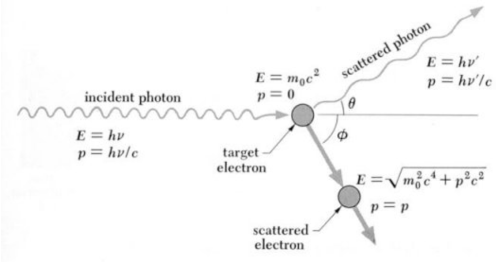


Figure 4: A visual representation of a Compton scattering of a photon off a recoil electron in a medium along with respective energies and scattering angles. Image from usr *ureview@bell.net.* Accessed 30/10/19 via `http://universe-review.ca/I15-72-Compton1.jpg`.

At energies higher than twice the rest mass of an electron pair production becomes energetically possible. In this case photon energy is used to create an electron-positron pair, with any excess shared between the two in the form of kinetic energy. Following the positron annihilation with electrons in the medium, radiation is produced in the form of two 511 keV photons that may superimpose on decay spectra of the daughter product, resulting in characteristic peaks at either or both 511 keV and 1.02 MeV (also called *escape peaks*). [29], [30]

2.1.2 X-Ray Attenuation

The penetrating ability of an X-ray beam is commonly measured in terms of the Half Value Layer (HVL). It is defined as the distance required to be traversed in a given material in order for the incident electric field to dissipate exactly half of its energy (beam intensity). The effective energy of a polychromatic X-ray beam is equivalent to that of a monoenergetic one possessing the same penetrating ability. The range of transmission is directly proportional to photon energy and inversely to the atomic number/density of the absorber material. It is quantified through the interaction probability per path length or linear attenuation coefficient μ:

$$\mu = \tau_{PE} + \sigma_{Compton} + \mu_{pair}, \tag{2}$$

which is simply the sum of all interaction probabilities. The transmitted intensity I can then be expressed in terms of incident beam intensity I_0 and material thickness t as [29]:

$$I = I_0 e^{-\mu t}. \tag{3}$$

2.1.3 X-Ray Production and Detection

Employing an X-ray tube is among the most common and straightforward methods of obtaining X-ray photons. It consists of an evacuated chamber containing a metal filament (cathode), which emits electrons upon heating (*thermionic emission*), and a high potential difference across the chamber that accelerates the electrons towards the anode. The bombarding electrons can then either decelerate due to Coulomb interactions with the electric field near the nucleus or knock out the electrons from the inner shell of the absorber material atoms. The

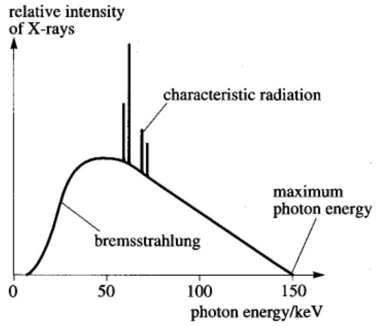


Figure 5: X-ray radiation emission spectrum. Image from Physics Open Lab. Accessed 30/10/19 via `http://physicsopenlab.org/wp-content/uploads/2017/08/graph-11.png`.

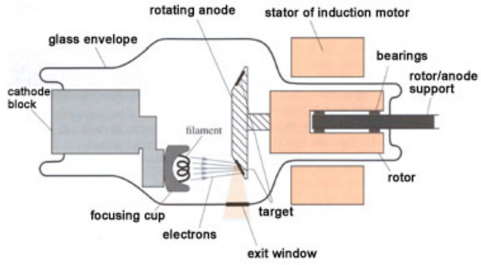


Figure 6: Schematic of a rotating anode X-ray tube cross-section. Image from Physics Open Lab. Accessed 30/10/19 via `https://www.sltinfo.com/wp-content/uploads/2016/04/x-ray-tube.jpg`.

former results in a continuous emission spectrum referred to as *bremsstrahlung*, while the latter gives rise to the pronounced peaks corresponding to the characteristic X-ray emissions due to inner shell transitions, as discussed prior in Section 2.1.1. An example of the resulting superimposed spectrum can be seen in Figure 5. Sometimes secondary emission of electrons may follow if sufficient amount of released energy is transferred into releasing them from the outer shell. These are referred to as *Auger* electrons and do not contribute to the process of X-ray production. The threshold on the intensities of the peaks (X-ray flux) is controlled by the number density of the electrons (tube current) and the K-edge of the filament material. This means that the delivered dose rate (see Section 2.1.5 below) increases linearly with tube current output. The maximum achievable emission energy is defined by the tube voltage applied. [24], [29] In some X-ray tubes the anode is rotated in order to prolong the lifetime of the material through redistributing the electron bombardment over its area. Figure 6 demonstrates the components of a rotating anode X-ray tube. The produced photons then pass through an exit window of a small $\mathbb{Z}$ material (usually beryllium or aluminium) to rule out the unwanted weak radiation that can be fully absorbed in the patient, causing damage and not contributing to image formation. Typical photon energies used in diagnostic radiology range from 20 keV to 150 keV (1 eV $\equiv 1.6 \times 10^{-19}$ Joules or kinetic energy gained by one electron through being accelerating from rest by a potential difference of 1 V in vacuum). [1], [29], [30] Due to the high flux of X-rays in CT tungsten anodes and cooling systems with water or oil are used to handle high temperatures and large heat dissipation. [25]

The detector quantum efficiency reflects the detected fraction of photons impinged on the surface and scales with the incident photon energy, absorber material and its thickness. Currently, solid-state detectors are used in CT due to their high DQE. They mostly consist of crystal or ceramic scintillators. When a valence electron in a scintillating material is excited by the incident X-ray photon, it jumps an energy band. Eventually the electron loses its energy through reemitting a visible light photons, which are then picked up by photodiodes or photocathodes of the photomultiplier tubes and transformed into an electrical current. [29], [30] This way, the output in CT is fully digitalised and available for evaluation within seconds.

2.1.4 Quantum Noise

There is a degree of uncertainty associated with quantum processes due to their inherently random nature. The statistical fluctuation on the number of detected photons obeys the Poisson distribution with relative counting error decreasing with the total number of photons N as $\propto \frac{\Delta N}{N} = \frac{\sqrt{N}}{N} = \frac{1}{\sqrt{N}}$. As the dose scales with the X-ray flux, at lower doses electronic noise becomes more significant. It normally manifests as a systematic error that is Gaussian-distributed and is usually caused by the equipment itself. Therefore, the total error on the signal due to noise ϵ_N will follow:

$$\epsilon_N \sim Poisson[N_0 \exp(-y)] + Gaussian[0, \sigma_e], \tag{4}$$

where N_0 is the incoming photon flux, y is the sinogram data (see definition in Section 2.3.2) and σ_e is the standard deviation of the electronic noise. While this representation of noise distribution is generally true for idealised X-ray measurements, it is not so straightforward in reconstructed CT images. [31]

2.1.5 Dose and Radiation Safety

Exposure to ionising radiation is linked to hazardous health implications that are either directly related to delivered dose (such as DNA damage and toxicity due to an increase in free radical density) or implicitly lead to long-term complications. Safety regulations are essential for minimising all radiation-associated risks for both the patients and the staff involved. Any delivered dose should be kept **A**s **L**ow **A**s **R**easonably **A**chievable (*ALARA*). Patient age is also a decisive factor because smaller organ sizes in younger patients deem them more vulnerable to radiation exposure and longer life expectancy allows for the detrimental effects to manifest. [32]

Energy deposition is typically measured in *Grays* (Gy), defined as the absorption of one Joule of radiation energy per kilogram of matter, while the biological effects due to received dose are quantified with *Sieverts* (Sv). Radiation exposure is quantified as the kinetic energy transfer to all secondary particles (ionised electrons and emission photons) by the primary flux photons: $K = \frac{\sum_i E_i}{dm}$, where m is the mass element where the incident energy is fully absorbed. K is measured in Gy and is called the *Kerma* (**K**inetic **E**nergy **R**eleased by uncharged particles per unit **M**ass). While Kerma deals with the associated energy transfer, the absorbed dose encapsulates all the energy that had been deposited in a given material. Naturally, its amount and the extent of physical and chemical changes scale with the type and strength of radiation, specific element half-life as well as ionisation density and scattering properties of a given material or metabolism in the body. Thus, the concept of the *effective dose* H_E had been introduced to estimate the health effects based on both the severity of radiation and specific organ sensitivity:

$$H_E = \sum_T \omega_T H_T, \tag{5}$$

where ω_T is tissue-specific weighting factor, $H_T = \sum_R \omega_R D_{T,R}$ is the *equivalent dose* absorbed by a given tissue type T and H_E, H_T are measured in Sv. The radiation equivalence weighting factor ω_R constitutes for the fact the extent of damage varies with the type of radiation, even if the absorbed dose is the same. For example, an absorbed photon of 1 mGy is equivalent to the dose of 1 mSv, while for 1 MeV neutron 1 mGy absorption results in a dose of 20 mSv ($\omega_R = 20$). [25], [33]

The effective dose upper limit as established by *ICRP* (International Commission on Radiological Protection) is approximately 1 mSv annually from occupational use for general public and up to 20 mSv for certified workers (delivered over a 5 year period). Typical CT scan effective doses vary from around 1 mSv (lung/head) to 25 mSv (whole body). [34] The exposure due to natural background radiation in Denmark constitutes 2 - 4 mSv per person per year. [35] The risk of developing cancer is estimated at 5% per Sv received (also subject to age group and body part exposed) and the lethal dose is defined as that causing death to 50% of exposed population in 30 days and ranges from 4 to 5 Sv (when received in a relatively small time period). [36]

2.2 Computed Tomography

Figure 7: A rendered image of a typical CT scanner from *SIEMENS*. Accessed 31/10/19 via: `https://www.siemens-healthineers.com`.

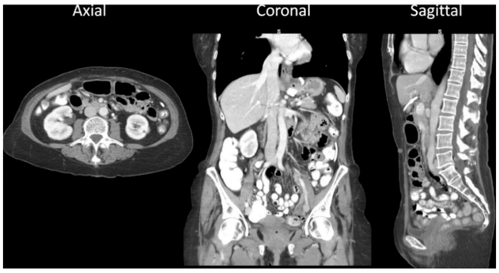


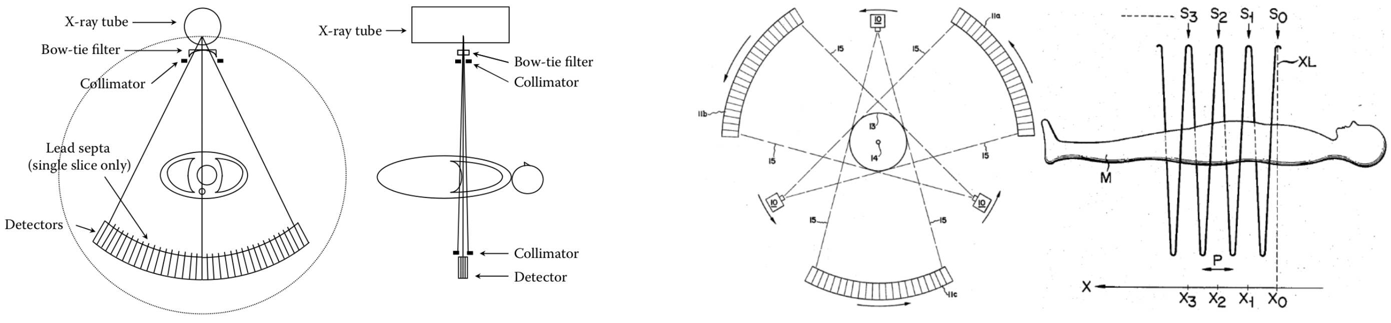
Figure 8: Reconstructed CT image examples in three standard slice orientations: axial, coronal and sagittal. Image from *Bushberg et al.* [27].

After its first clinical implementation in the early 1970s, the CT scanning method was subsequently recognised with the 1979 Nobel Prize in Medicine jointly shared between Godfrey N. Hounsfield and Allan M. Cormack. [37] Presently, the refinements in the CT imaging setup had resulted in high spatial and contrast resolution with a large field of view and acquisition times of mere seconds.

A radiographic image is a two-dimensional projection of a three-dimensional distribution of X-ray paths and associated attenuating properties along those paths. The image then represents a cross section containing one-dimensional line projections from different angles. It is possible to reconstruct the cross-sectional image from a given set of ”raw” projection data by taking an inverse *Radon transform* (see Section 2.3.2) and the volumetric object is formed by building a stack of reconstructed images. A CT scanner acquires the projection data in steps by rotating an X-ray (along with detector array) source around the patient and moving the table across after each exposure. Figures 7 and 8 show a visual representation of a typical CT scanner and formed CT images respectively.

2.2.1 Instrumentation

This section gives an brief overview of the fundamental components of a generic CT scanner. As previously seen in Figure 7, a CT scanner consists of a movable patient table and a cylindrical casing that houses a fast-rotating gantry upon which both the X-ray tube and the detector array are mounted. This setup is demonstrated schematically in Figure 9a. The X-ray attenuation is then measured along a line between the source and a given detector. Modern scanners also utilise spiral/helical scanning as well as multiple detector arrays to reduce acquisition time (Figure 9b). Moreover, in such a setup the detectors outside of the *field of view* (FOV) of the beam can be used to measure the unattenuated beam intensity (making it, thus, possible to easily infer μ).



(a)

(b)

Figure 9: Schematic representation of basic CT components and scanning methods. (a) A gantry with multiple detector rows as seen from patient’s feet and side. Collimation and filter systems are employed to focus the X-ray beam and modulate its intensity, while lead septa isolate signals from neighbouring detectors. (b) The concepts of multi-detector/source (left) and helical scanning (right). The former is able to acquire multiple slices simultaneously, while the latter rotates the source around the isocentre while the patient table is moved across. The resulting X-ray tube path about the patient is then a helix of pitch $p = \frac{X}{S}$, where x is the bed advancement per source rotation and s is the beam collimation. It follows that lower pitch leads to higher absorbed dose. Hence, tube current modulation is required to avoid unnecessary exposure, while maintaining image quality irrespective of the pitch. Images from of *Flower* (2012) [1] and *Dance et al.* (2014) [25].

2.2.2 Transmission Profile, Hounsfield Units and Image Formation

As the source is moved across the patient in N steps, X-ray transmissions are measured for each cross-section at N angles. Every slice is then divided into $N \times N$ pixels and the total attenuation per slice is measured as a sum of all pixel contributions along a given X-ray path l as a line integral:

$$\int \mu(x, y)dl = ln[\frac{I_0}{I}] = \sum_{i,j}^{N} x_{pq,\{i,j\}}\mu_{i,j} = \Lambda_{pq}, \tag{6}$$

where i, j is a given pixel position, pq are given measurement angle and offset respectively, x is the length of attenuation line in pixel i, j, such that $x_{pq,ij}$ is a contribution of pixel $\{i, j\}$ to a measurement pq (fraction of pixel $\{i, j\}$ intercepting a given X-ray) and μ, I, I_0 have their previous meanings. Thus, a set of N parallel projections $\Lambda_\theta(x')$ is obtained for each measurement angle. This results in a matrix of linear attenuation coefficients inferred from the respective degree of attenuation of the incident beam in a given pixel/voxel.

In CT imaging given μ values are translated into corresponding CT numbers, measured in Hounsfield units (HU) relative to the linear attenuation coefficient of water at room temperature [25]:

$$HU_{material} = \frac{\mu_{material} - \mu_{H_2O}}{\mu_{H_2O} - \mu_{air}} \times 1000. \tag{7}$$

It follows that HU$_{H_2O}$ = 0 and HU$_{air}$ = -1000. A change of 1 to the HU value translates to the 0.1% change in the μ of water and, thus, reflects the relative difference between the linear attenuation coefficient of a given material and that of water. The Hounsfield scale was introduced for establishing practical reference values for anatomical tissue types.

2.3 Image Reconstruction

This section considers some of the mathematical concepts of signal processing and standard reconstruction algorithms associated with CT data, namely the *Filtered Backprojection* and *Iterative Reconstruction.*

2.3.1 Signal Processing

In reality, a response of an imaging system is not entirely isotropic, but has an inherent degree of blurring associated with the detector output. The *Point Spread Function* (PSF) describes the blurring on a point source mathematically. An image can then be formed by convolving (see more on convolution in Section 2.6) a 2D object profile with its respective PSF. The added PSF blurring contributions overlap more in the vicinity of the object position, thus, forming a region of highest intensity in the final image. An alternative description of the blurring of an imaging system is provided by the *Modulation Transfer Function* (MTF). MTF is a multiplicative factor reflective of the "degree of blurring" (reduction in the signal intensity). For a sinusoidal signal of constant amplitude, the effects of blurring will be more prominent at higher spatial frequencies and result in lower output contrast. It follows that the image can be built by decomposing an object into superposition of spatial frequencies and multiplying them by an appropriate MTF (Figure 10). From the *Convolution Theorem* that states that the Fourier Transform (FT) of the convolution of two functions is equal to the product of their FTs it follows that MTF is the FT of the PSF: $MTF(x,y) = |FT[PSF(x,y)]|$. [38] [39]

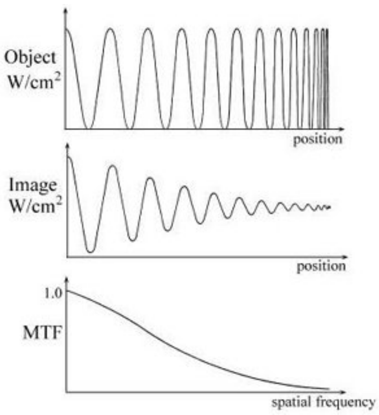


Figure 10: Signal (top), imaging system output (centre) and respective MTF that defines resolution capabilities of the system (bottom).

2.3.2 Filtered Backprojection

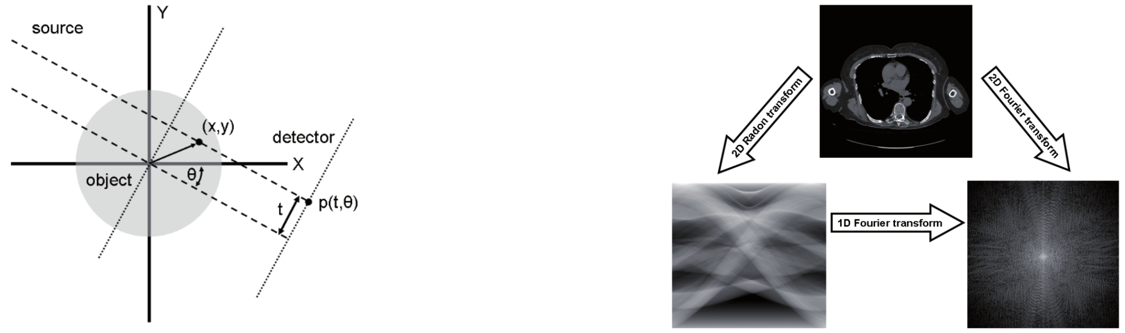
The backprojection method is based on the intuitive concept of taking the transmission profiles, measured in accordance with Equation (6), at specific angles and distributing that signal uniformly back at the same angle as the respective projection.

Projections of a 2D image in Cartesian space (x, y) with parallel rays yield a single line in the projection space (also called the *Radon* space). Projection $p(t, \theta)$ is then a function of the distance of a given ray from the object's isocentre t and the projection angle θ (see Figure 11a). The raw data in CT is represented by a complete set of profiles (all measured projections at each angles) that form a 2D Radon space or a *sinogram* (an example plot of which can be found in Figure 11b). The *Central Slice Theorem* states that a Fourier Transform of a given $p(t, \theta)$ translates to a line in the 2D Fourier space (also *k-space*), angled at θ. This operation is visualised in Figure 11b), where it can also be seen that the transformed projections in the frequency domain radiate outwards. This leads to over-representation of lower spatial frequencies in the final image and inevitable blurring. Instead of applying weights to individual frequencies and handling the cumbersome inverse 2D FT, the filtered backprojection approach is used instead. Here, each projection is convolved with a filter that suppresses lower frequencies and amplifies the higher ones responsible for the edge behavior in the Fourier space prior to backprojecting. An example of a typical filter and its effect on the reconstructed image resolution is shown in Figure 12. Figure 13 illustrates how the backprojections at different angles from Radon space with a filter applied in the Fourier domain form an image. Overall, the reconstructed image $f(x, y)$ can be expressed as:

$$f(x,y) = \int_0^{\pi} d\theta \int_{-\infty}^{+\infty} p(k,\theta)|k|e^{i2\pi kt}dk, \tag{8}$$

where $p(k, \theta)$ is the 1D FT of the 1D projection at an angle θ (typically projections measured from 0 to π) and $|k|$ is a filter in the frequency domain. [27], [38], [39]

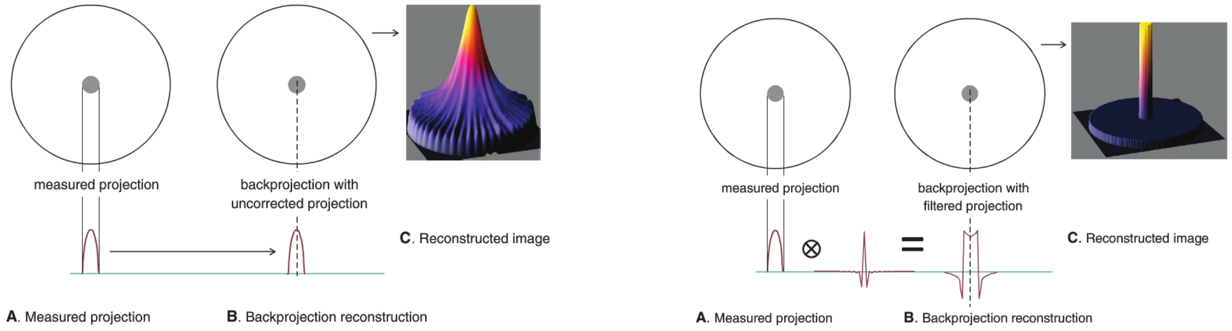
Different reconstruction filters or *kernels* are available in CT protocols depending on the clinical purpose. These include smoothing filters for noise reduction or edge/contrast enhancing convolution kernels.



(a) Single transmission profile geometry. An X-ray is transmitted through the point (x, y) in a 2D object, at a distance t from its centre and at an angle θ with respect to the detector. For each set of transmissions at each given angle there exists a 1D line in the Radon space.

(b) The mathematical domains and respective transformations involved in FBP. The image slice (top) and corresponding projection space (bottom left). When a single point profile is measured at different angles, a sine wave is formed. A sinogram is therefore a collection of such waves for every projection point in the image domain. It follows that in practice one can obtain an image from a given sinogram by performing a 2D inverse Radon transform. A *Fourier map* of the image is then built from 2D arrays of FT of each projection in the sinogram at their respective angles, in accordance with the Central Slice Theorem.

Figure 11: Geometrical considerations of backprojection. Images courtesy of *Dance et al.* [25].


measured projection
backprojection with uncorrected projection
C. Reconstructed image
A. Measured projection
B. Backprojection reconstruction
measured projection
backprojection with filtered projection
⊗
=
C. Reconstructed image
A. Measured projection
B. Backprojection reconstruction

(a)

(b)

Figure 12: Backprojecting a measured intensity profile with (a) and without a filter (b). Convolving the projection with a filter results in a significantly higher spatial resolution. Images courtesy of *Bushberg et al.* [27].

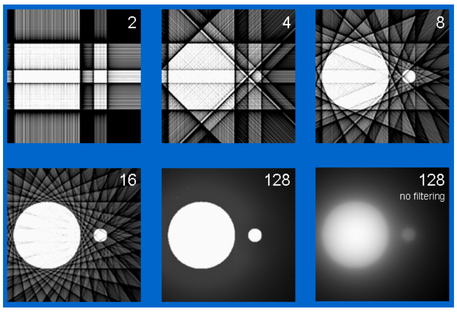


Figure 13: The effect of the number of used projections (2, 4, 8, 16 and 128) in FBP on the reconstructed images. Bottom right shows the result for backprojection without prior filtering. Image courtesy of *ImPACT*, UK. Accessed: 21/01/19 via `http://www.impactscan.org/slides/impactcourse/basic_principles_of_ct/image15.html`.

2.3.3 Iterative Reconstruction

Due to the rapid advancement in computer hardware and algorithm architectures, iterative reconstruction techniques had been made possible and are widely utilised. The iterative reconstruction method replaces the line integrals for the projections with a set of linear equations to be solved:

$$\hat{\lambda}(\phi, x') = \sum_{i=1}^{I} \alpha_i(\phi, x')\mu_i, \tag{9}$$

where $\hat{\lambda}$ is the computed projection, $\alpha_i(\phi, x')$ is the path length of projection (ϕ, x') in the pixel i and the rest have their previous meanings. The values of μ_i are computed numerically until $\hat{\lambda}$ resembles measured projection λ. The calculation methods vary across algorithms. A simplified example is shown in Figure 14. [1], [38]

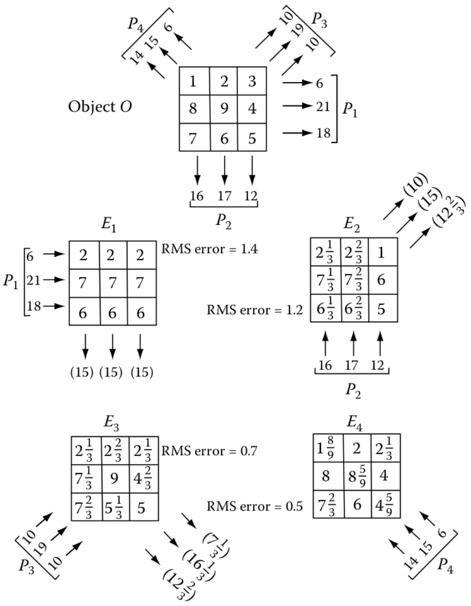

P_4 14 15 6
P_3 10 19 10
Object O
1 2 3
8 9 4
7 6 5
6 21 18 P_1
16 17 12
P_2
E_1
P_1 6 21 18
2 2 2
7 7 7
6 6 6
(15) (15) (15)
RMS error = 1.4
E_2
(10) (15) $(12\frac{2}{3})$
$2\frac{1}{3}$ $2\frac{2}{3}$ 1
$7\frac{1}{3}$ $7\frac{2}{3}$ 6
$6\frac{1}{3}$ $6\frac{2}{3}$ 5
RMS error = 1.2
16 17 12
P_2
E_3
$2\frac{1}{3}$ $2\frac{2}{3}$ $2\frac{1}{3}$
$7\frac{1}{3}$ 9 $4\frac{2}{3}$
$7\frac{2}{3}$ $5\frac{1}{3}$ 5
RMS error = 0.7
10 19 10 P_3
$(7\frac{1}{3})$ $(16\frac{1}{3})$ $(12\frac{2}{3})$
E_4
$1\frac{8}{9}$ 2 $2\frac{1}{3}$
8 $8\frac{5}{9}$ 4
$7\frac{2}{3}$ 6 $4\frac{5}{9}$
RMS error = 0.5
14 15 6 P_4

Figure 14: A schematic representation of iterative reconstruction. Three-ray projections $\mathbf{P_{1-4}}$ are taken through the 9-pixel object $\mathbf{O}$ additively. The projection dataset is then sequentially compared with respective estimate $\mathbf{E_{1-4}}$ grids and the RMS difference between them is used to correct and backproject the updated image information. This continues until a stopping criterion like sufficient RMS or iteration number is reached. Image courtesy of *Flower* (2012) [1].

2.4 Image Quality

There are many factors that contribute to a reconstructed image appearance and its diagnostic value. Among the most important qualities in CT is the *contrast* or the ability to distinguish small differences in greyscale values between neighbouring image regions. Even though a lower tube voltage will result in better contrast, it is typically kept relatively high to sustain good X-Ray transmission. Achieving acceptable contrast becomes progressively more tricky for smaller structures in lower contrast regions since they are usually obstructed by noise. As previously established, this noise is largely caused by the insufficient number of photons reaching the detector (Section 2.1.4), but simply raising the tube current (mAs) comes at a trade-off of patient exposure. Alternatively, noise can be reduced through increasing slice thickness or employing smoothing filters at the cost of degrading the spatial resolution.

2.4.1 CT artefacts

The causes of artefacts in CT can be broadly divided into three categories: physics, hardware and patient. All three will be be briefly discussed below in given order.

Aliasing refers to undersampling in the number of projections and ray paths per projection. Moreover, there is a fundamental limitation on the highest spatial frequency that can be discretely sampled. If that frequency is higher than the *Nyquist frequency*, defined as $f_N = \frac{1}{2}\upsilon$, where υ is the spatial sampling rate, the true signal cannot be recorded unambiguously. Finally, the sampling resolution is physically limited by the detector width itself. Aliasing leads to higher frequency signals appearing as lower frequency artefacts (Section 2.3.2) and creates “streaks” in the reconstructed image. **Beam hardening** occurs when lower energy X-rays in a polychromatic beam get attenuated in the body more, resulting in overall effective energy increase in the spectrum. As a result, the “hardened” beam will appear brighter at the tissue exit point as compared to the entry due to higher energy photons being attenuated less. This leads to underestimation of the linear attenuation coefficient and has to be corrected for (e.g. with post-processing techniques or dual-energy scans). Another potential cause for CT number underestimation is the **partial volume averaging**, appearing when the scanned object does not fill the scan plane.

Ring artefacts can appear due to system miscalibration or malfunctioning detector elements and *arc* artefacts are a consequence of short-circuit within the tube that leads to loss in X-ray information.

Finally, artefacts can be caused by patient **motion** during the scan or the presence of a highly attenuating material, like a **metal** implant. [25], [27], [38], [39]

* * *

2.5 Deep Learning

In the recent years Machine Learning (ML) has gained great prominence across various research disciplines, resulting in a plethora of scientific advances from object recognition [40] and speech synthesis [41] to abnormal pattern detection [42] and self-operating machinery [43]. Specifically, Deep Learning, originating from the early 1980s [44], has become an asset in the field of medical imaging due to its numerous successful applications in image-to-image translation tasks [45], including but not limited to segmentation and classification [46], augmentation [47], synthesis [48] and super-resolution [49]. Among the most valuable for diagnostic purposes are the image reconstruction methods: from artefacts correction [50] to denoising. These techniques aim to assist the qualified professionals that presently act as the gold standard for medical image evaluation.

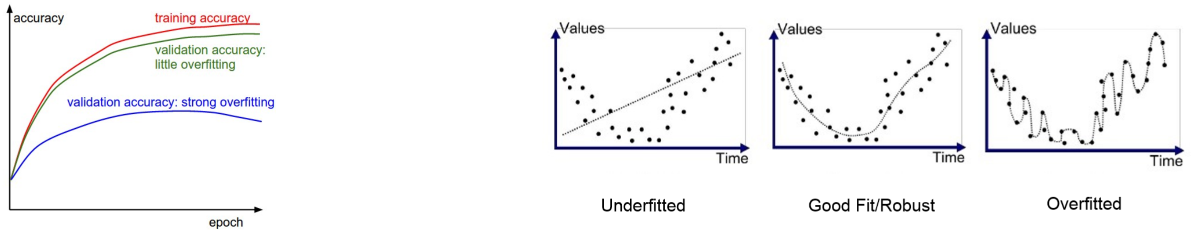
2.5.1 The Basics of Machine Learning

The basic dynamics behind a ML algorithm comprises measured data of one or more **features** (e.g. image pixels or edges), a model that inputs data and carries out a task and its task-specific performance measure. A program is said to "learn" from a given example if doing so results in improved model performance. [51] The accuracy of the model is calculated with an **error rate**, reflective of the proportion of data samples or average associated probabilities for which the model produces an undesirable result.

The degree of change to the model in response to error estimations during training is referred to as **learning rate**. Learning rate is an example of a collection of ML algorithm **hyperparameters**, which are values set before the training process (for instance, number of hidden layers, kernel sizes, patch sizes). If the learning rate is too fast, the model converges too quickly, producing a suboptimal result or divergence, whereas a learning rate that is too small can take too long to train or even get stuck.

Machine learning can be divided into two broad categories: **supervised**, where each feature set is also associated with a specific label that the algorithm learns to predict from a given example, and **unsupervised**. The latter method typically learns how to predict a distribution that generates the sample set either explicitly or implicitly by using a dataset with many features, such is the case in image denoising.

Training a model involves sampling from this data-generating distribution and separating a given example into data used for training and fitting the model (**training set**) as well as "unseen" data for which the the predictions are to be made (**test set**). Test data is only used to assess the performance. The degree to which a given model performs as expected on the previously unseen data is referred to as **generalisation** or **test error**. The goal is then to minimise the training error (see more in Section 2.5.4), while also reducing the gap between training and testing error. A situation where a model is not able to reach an acceptably low training error is called **underfitting**, while the gap between testing and training error that is too large is referred to as **overfitting**. Figure 15a shows a schematic plot of model fit accuracy as a function of **epochs** (number of times that a given neural network had iterated over the entire training data) comparing strong and little overfitting as well as some visual examples. The ability of a model to find appropriate fits for both the training and the test data is called the **capacity**. It should be defined by the complexity of the task at hand, as setting the capacity higher than necessary can lead to overfitting. [52]



(a)

(b)

Figure 15: A trained model is validated against test data. (a) When training fit accuracy (red) approximately follows the validation curve, little to no overfitting is assumed (green). Blue curve represents an overfitted model. (b) An underfitted model fails to learn, while an overfitted one is too dependent on the training data. Image courtesy of user cs231n (karpathy@cs.stanford.edu). Accessed 23/10/19 via `http://cs231n.github.io/assets/nn3/accuracies.jpeg`.

Commonly a given dataset is represented in terms of a matrix, also called **design matrix**, (or a set of example vectors when dealing with heterogeneous data), where each row contains an example and each column an individual feature. Supervised learning also requires supplementary label vector(s). An ML algorithm then acts as a prescription for transforming a given design matrix vector into predicted output.

2.5.2 Deep Neural Network

Let an $n \times m$ matrix of pixel values be an input fed to a neural network of several **layers** (described below) and subsequently passed forward through each layers. Because only the input/output information is available to the user, the layers are considered "hidden", while their number defines the "depth" of the neural network. Each layer consists of individual units called **neurons**, all of which are connected to the neurons in the following layer (also referred to as *fully-connected*). Figure 16 is a simplified example diagram of a neural network with two hidden layers and respective interconnections of nodes. Deep neural network is a collective term that encompasses feed-forward networks with many hidden layers.

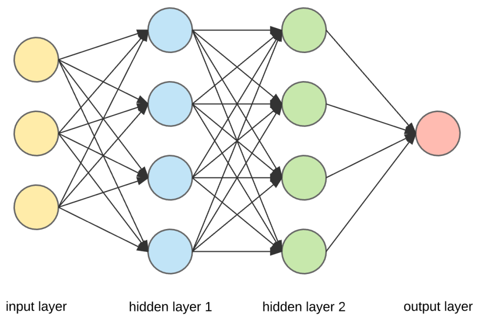


Figure 16: A simple neural network containing two hidden layers of neurons. The lines represent the information passed from a given neuron to all the neurons in the next layer. The last layer transforms the learned information into an output that is compared with the target value. Image from Sorokina, K. (2017). Accessed 15/10/19 via `https://miro.medium.com/max/1318/1*3fA77_mLNiJTSgZFhYnU0Q@2x.png`.

2.5.3 The Activation Function

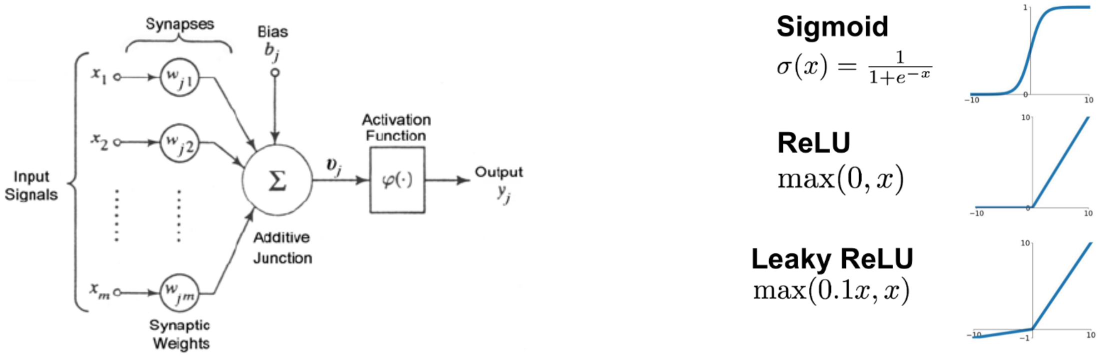
What separates a deep learning algorithm from a simple regression optimisation is the ability to handle non-linear functions and understanding the interactions between all the input variables. This is made possible via the **activation function** that is responsible for transforming the weighted input sum into corresponding output (the process of **neuron activation**, depicted schematically in Figure 17a). The neurons are then activated when the weighted sum exceeds a specific "activation number" a, defined as follows:

$$a^{(L)} = \sigma[w^{(L)}a^{(L-1)} + b^{(L)}] = \sigma z^{(L)}, \tag{10}$$

for a single neuron located in a layer L, where $a^{(L-1)}$ is the activation of a neuron in the previous layer, w is the weighting term and b is the bias, which is an independent parameter for each layer introduced to ensure the model fits need not necessarily pass through the origin. Thus, in the absence of an input, the layer output would be biased towards b. σ represents a non-linear function that forces a given value into a desired range, typically used for normalisation and computational efficiency (see Figure 17b for examples). The relevant weighted sum is expressed in terms of $z^{(L)}$ for simplification purposes. Since there is usually a number of neurons in a given layer (as is shown in Figure 16), w is not a parameter, but rather a vector of parameters that determines the strength of node interconnections or, in other words, how each feature affects the mapping from parameters to predictions. Therefore, for an $n \times m$ matrix it is the weighted sum of respective activations:

$$a_m^{(L)} = \sigma \sum_{n,m=1}^{n_{L-1}} w_{n,m}^{(L)} a_{m-1}^{(L-1)} + b_m^{(L)}, \ L = \{0, 1, ..., n\}; \tag{11}$$

that determines the likelihood of activation of a single neuron in a given layer L to result in an overall algorithm output value getting closer to the desired output. Naturally, a positive weight increases a specific neuron's associated prediction value, while a negative one decreases it.



(a) Neuron activation mechanism. A given activation function determines which neurons are fed as the input to the next layer in the form of their weighted sum and corresponding biases. Image from *Guarnieri et al.* [53].

(b) The *sigmoid* function transforms real numbers into outputs within the $[0, 1]$ range. The *ReLU* is zero for all inputs $x < 0$ and linear otherwise. Modified image from *Jain (2019)* [54].

Figure 17: (a) A given neuron activation is triggered when the weighted sum of inputs and biases exceeds the threshold. (b) Plot of some common activation functions.

ReLU stands for *Rectified Linear Unit* and is an activation function, responsible for transforming all the given node inputs to values in the range $(0, x)$. More specifically, ReLU outputs 0 for all negative inputs and simply returns all the non-negative values. Such a normalisation process eliminates all the negative pixel values and introduces non-linearity after affine convolutional transformations. Applying ReLU also assists gradient learning as the second derivative of the rectifier is almost always 0 and, thus, all second order effects become negligible. Often, a *Leaky ReLU* is used instead, where all the negative inputs are transformed with some small near-zero slope coefficients to ensure activation by most inputs from the training data and, thus, a better estimation of all the gradient contributions. [52].

2.5.4 Loss Functions and Optimisation

Optimising a deep learning algorithm implies indirectly improving model performance on a given training example, estimated in terms of a **loss function**. One of the most straightforward approaches to quantifying the degree of its accuracy is to calculate the mean squared error (MSE) of the model on the test set. The single resulting value C_0 is then the *cost* of learning from a given example:

$$C_0 = (\hat{y} - y)^2, \tag{12}$$

where $\hat{y}$ is the network prediction and y is the desired output. Following this logic, the magnitude of the activation change of a given neuron, determined by the cumulative weights from the previous layer, is directly proportional to how close it is to the expected value. The primary goal of a neural network is to find the most effective way to adjust weights and biases in order to minimise the cost of training. In mathematical terms, one needs to find the most rapid **gradient descent** (of the training error) on the cost function across all training data:

$$-\nabla \mathbf{C}(\tilde{\mathbf{w}}, \tilde{\mathbf{b}}) = \frac{\partial C}{\partial w^{(L)}} + \frac{\partial C}{\partial b^{(L)}} = \frac{1}{n}\sum_{i=0}^{n_L-1} \frac{\partial C_i}{\partial w^{(L)}_{n,m}} + \frac{\partial C_i}{\partial z^{(L)}} = \mathbf{0}; \quad \vec{w} = \begin{bmatrix} w^{(0)} \\ \vdots \\ w^{(L)} \end{bmatrix}, \vec{b} = \begin{bmatrix} b^{(0)} \\ \vdots \\ b^{(L)} \end{bmatrix}. \tag{13}$$

In other words, the local minima are found by calculating the gradient loss for the training dataset and updating the parameters that point in the opposite direction until a local minimum is found.

It is important to note that the notions of cost and loss functions are interchangeable across literature and publications. The main purpose, however, is to establish a network optimisation tool in the form of a function to be minimised.

Typically one deals with either a numerical quantity or a label as the model output. Regression loss functions handle the first category. Approaches other than MSE in Equation (12), also known as *L2 Loss*, can be utilised for loss minimisation. For instance, the sum of absolute differences between target and prediction values can act as an alternative loss function. This Mean Absolute Error (MAE) or *L1 Loss*, defined $\frac{1}{N}\sum_{i=0}^{N}\|a_i^{(L)}-y_i\|$ accordingly, is less optimal for training (see Figure 18), but is more responsive to statistical outliers in the training data. [52]

2.5.5 Backpropagation

The **backpropagation** algorithm is responsible for determining how sensitive the cost function is to single changes in weights and biases for every training example. Firstly, the components that minimise the cost of network training are determined by applying chain rule to the cost function:

$$\frac{\partial C_i}{\partial w_{n,m}^{(L)}} = \frac{\partial z_m^{(L)}}{\partial w_{n,m}^{(L)}} \frac{\partial a_m^{(L)}}{\partial z_m^{(L)}} \frac{\partial C_i}{\partial a_m^{(L)}}. \tag{14}$$

One can then infer from the term $\frac{\partial a_m^{(L)}}{\partial w_{n,m}^{L}}$ that the strength of activation of node m in layer L depends on activation of node n in layer $L-1$, determined by weight $w_{n,m}^{L}$, in accordance with Equation (11). Finally, the response of cost function to this preceding activation $a_n^{(L-1)}$ can be expressed in a similar way to expansion in Equation (14):

$$\frac{\partial C_i}{\partial a_n^{(L-1)}} = \frac{\partial z_m^{(L)}}{\partial a_n^{(L-1)}} \frac{\partial a_m^{(L)}}{\partial z_m^{(L)}} \frac{\partial C_i}{\partial a_m^{(L)}}. \tag{15}$$

This demonstrates that the loss in one given layer $L-1$ is directly affected by the loss in the next layer L, encompassing the notion of backpropagation, where "learned" weights and biases in the second to last layers are used to update the previous layers recursively until the optimal fit is found. Initialising weights and biases is of great significance to any training process as it determines how close the model is from a given target at the start of the optimisation process. The parameters can either be set to zero, drawn randomly or obtained from a similar model that had been previously trained in the process known as *transfer learning.* [55]

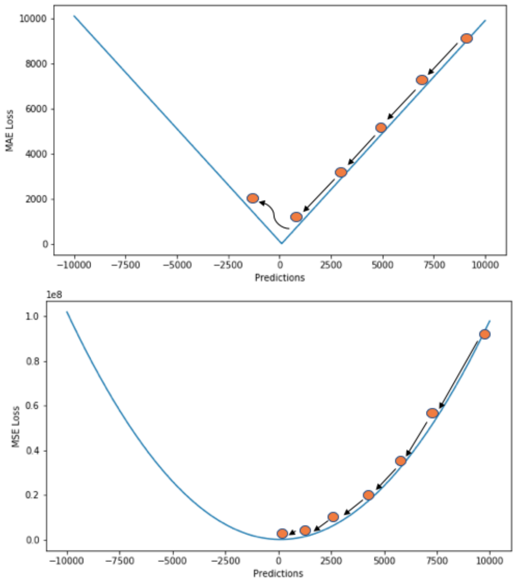


Figure 18: MAE loss (top) and MSE loss (bottom) plots for an example where the true target value is 100 and the predicted values range from -10000 to 10000. The gradient of MAE loss is constant due to the function's linear nature, while MSE gradient decreases as loss approaches 0, resulting in better convergence. Image courtesy of Prince Grover (2018), accessed 09/04/19 via: `https://heartbeat.fritz.ai`.

The above considerations scrutinise a generalised cost function before applying any non-linear activation function. Moreover, computing the negative gradient across all data is too computationally cumbersome in reality. Instead, one would divide the data into *batches* and calculate the corresponding non-linear *stochastic* gradient descent instead.

Batch size is a hyperparameter that defines the number of samples iterated over before applying backpropagation.

There also exist auxiliary strategies aimed to optimise gradient descent. One example is the *Adam* [56] optimisation algorithm that utilises the *adaptive moment* method to continuously adjust individual learning rates for each parameter involved. The notion of the so-called momentum is inspired by the physical analogy and is estimated based on the cumulative exponentially decaying average of past gradients for each training batch.

In summary, basic feed-forward (deep) neural networks (also called *multilayer perceptrons* or *vanilla* networks) have an input layer, followed by hidden and output layers consisting of nodes with associated non-linear activation function. A network is trained to learn a non-linear mapping from a collection of examples to find a model approximation of some function of interest that would best describe both test and training data distributions using backpropagation to update the model parameters and iterative optimisation algorithms to minimise the training error gradient.

2.5.6 Model Regularisation

Simultaneously, various model regularisation strategies are typically employed during training to achieve the most efficient model performance on a given test set, while addressing problems like overfitting. For example, by limiting a model's capacity through introducing weight penalties (either through adding sparsity or decay of the large components in the weight matrix, know as $\mathbf{L}^1$ and $\mathbf{L}^2$ **regularisations** respectively). The technique of **batch normalisation** standardises each "mini-batch" input by calculating the mean and standard deviation per given batch to stabilise the training. When training a neural network with many layers, **droupout** or **long-skip connection** methods can be used to reduce the computational toll. The former intentionally ignores several random layer outputs to create an effect of fewer nodes and connections, while the latter skips layers when feeding information forward. **Early stopping** is used for models, whose validation errors begin to increase past a certain time. [52]

2.6 Convolutional Neural Networks

$$(f * g)(t) = \int_{-\infty}^{\infty} d\tau f(t-\tau)g(\tau). \tag{16}$$

Equation (16) is a general mathematical definition of the **convolution** operation acting on two functions $f(\tau)$ and $g(\tau)$, where the integral represents the amount of the overlap as g is shifted over f. In the context of image processing one is dealing with a finite set of digital measurements sampled from some analog signal. A measurement is, thus, a weighted average of these samples or a convolution of the sampling distribution with a weighting function. In neural network terminology f is a multidimensional input data array, g is is a multidimensional parameter array or **kernel** and the output is referred to as **feature map**.

A vanilla network acts similarly to a matrix multiplication of data and parameters. Its application is straight-forward for handling data of same dimensionality. Convolutional Neural Networks (CNN), on the other hand, are not constrained to the entire data set or same-size inputs, but utilise sparse representation, storing the important parameters only (e.g. edges) and omitting the rest. Using smaller kernels also reduces the computational complexity. Figure 19 is a simplistic visualisation of an **image patch** convolution with a filter (a bunch of features stored by the kernel). Patch size is a hyperparameter that defines the pixel area seen by the kernel at any given time. The convolution filter is moved across the input image patch in a number of steps. Every step, all the lined up image and feature pixel values are multiplied and divided by the total number of pixels in the kernel, forming a corresponding single pixel value in the output filtered image. A stack of filtered images forms a **convolution layer**. [57] A typical convolutional neural network layer consists of several suboperations: convolutions with associated linear activations (as discussed in Section 2.5.3), followed by a non-linear activation (ReLU) and a **pooling function**. In the process of pooling, a network designer picks a window (kernel) size and the length of **strides**, with which it is "walked" across the image to be filtered. This is done to reduced dimensionality and introduce high order features. [52]

2.7 Generative Adversarial Networks

Generative network types are useful, among other things, for manipulating high-dimensional probability distributions and predicting missing data inputs. [59] Generative Adversarial Network (GAN) is a neural network type that works on the basis of two networks being trained against each other. One is responsible for synthesising samples that mimic the training data distribution, while the other differentiates between real and generated samples. Since the introduction of its concept in 2014 (*Goodfellow et al.* [60]) GAN has found numerous successful applications in the medical imaging field across various tasks and modalities. [61]

In a GAN framework the generator machine G samples data distribution p_x to learn a mapping from random noise to data space (distributed with some unknown p_{data}) through implicitly defined generative distribution p_g over sample space x. The generator is usually conditioned upon auxiliary information (also referred to as

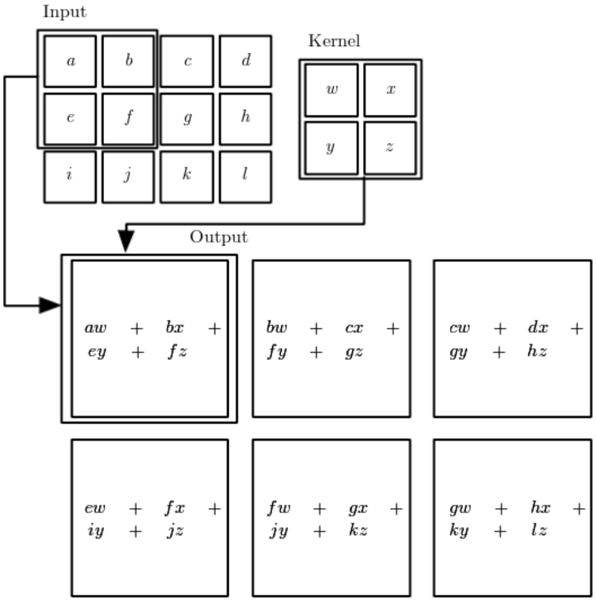


Figure 19: The squares represent pixel values stored in the data matrix. Each pixel in the output image is formed by convolving input image pixels with respective kernel pixels. The resulting pixel value is then the average of all corresponding image and feature pixel values, obtained through element-wise multiplication. The filtered image is formed by moving the filter across the selected patch. Image from of *Goodfellow et al.* (2016) [52].

conditional GAN or simply *cGAN*), for example an LDCT image to be denoised. The discriminator network D model parameters are simultaneously updated until $\frac{p_{data}(x)}{p_{data}(x)+p_g(x)}$ in its algorithm converges to $\frac{1}{2}$, effectively reaching the point at which $p_{data} = p_g$. The training itself is performed alternately, keeping the discriminator constant such that the generator has a fixed learning target during its training phase. Similarly, the generator is held constant during discriminator training, because the latter is forced to find more intricate solutions each time the generator improves its performance. Thus, the process of assessing a trained and an untrained generator is inherently different.

Generally speaking, GAN networks aim to solve a min/max optimisation problem through finding optimal parameters for the two (or more) networks in order to reach an equilibrium where neither of them can further reduce the cost function. It then follows that the loss for a GAN, conditioned upon a real data example y, can be quantified as:

$$\min_G \max_D V(G, D) = \mathbb{E}_{x,y\sim p_{data}(x,y)}[\log(D(x, y)] + \mathbb{E}_{x\sim p_x(x),z\sim p_z(z)}[\log(1 - D(G(x, z), x))], \tag{17}$$

where $V(G, D)$ is the overall cost function and $p_z(z)$ is the (random) noise distribution. In simple terms, training a conditional GAN to achieve the optimal algorithm convergence to $p_{data} = p_g$ involves sampling mini-batches of noise $\{z_i\}, i = 1, ..., m$ and feature vectors $\{x_i\}$ and computing the ascending gradient of the discriminator with the guidance of the target examples $\{y_i\}$:

$$\nabla_{\lambda_d} \frac{1}{m} \sum_{i=1}^{m} [log(D(x_i, y_i)) + log(1 - D(G(x_i, z_i), x_i))], \tag{18}$$

where λ_d represents the discriminator parameters and $D(x, y)$, $D(G(x, z), x)$ are the probabilities of a given sample coming from the data, p_{data}, or generated, p_g, distributions respectively. Once the discriminator is optimised accordingly, the error is backpropagated to update the generator. For smaller datasets D should be optimised in several steps per each training iteration to avoid overfitting. G is trained to minimise the probability of D assigning the correct label to a given sample x. This is achieved through maximising $log(D(G(x, z), x))$ in a way guided by the loss calculated with the feedback from D (a more comprehensive mathematical formulation of the above ideas can be found in *Goodfellow et al.* (2014) [60]). This approach had been empirically proven to be more robust than minimising $log(1-D(G(x,z),x))$, following the logic of Equation (18). A more stable learning is achieved by G maximising the log-probability of D being wrong, because at initial stages of the training G produces poor output which D can confidently reject, resulting in an overall saturation of log-probability. [59] In a conditional GAN, typically some form of an MSE loss is also introduced to minimise the difference between generated, $G(x)$, and real samples, y. [6] A schematic example of the above training dynamics can be found in Figure 20.

It is believed that GANs are, on average, a success in producing realistic looking images due to moving from traditional *max-likelihood.* [63] Naturally, variations in the vanilla GAN architecture, as well as tailored features to address specific challenges in the medical imaging field, had been proposed in the recently published works ([7]-[5], [19]).

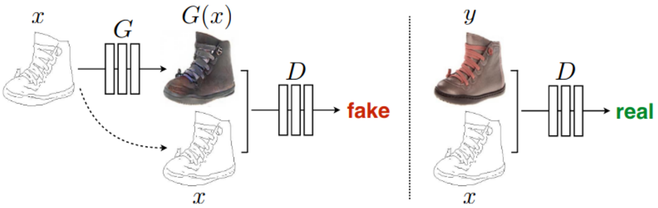


Figure 20: A schematic of the conditional GAN training dynamics based on the image domain adaptation example. The generator G is fed a variable x which it then maps to $G(x)$. The aim of G is to convince D that generated sample $G(x)$ had been drawn from the real data sample (like training example y). The discriminator is also conditioned upon the input to G, such that it effectively differentiates between real and fake image *pairs.* It aims for $D(y)$ to approach 1 and $D(G(x))$ to approach 0. The generator is the sufficiently optimised when $D(x) = \frac{1}{2}$ for $\forall x$. Image from of *Isola et al.* [62].

2.7.1 SAGAN

An example of a more complex GAN is an adversarial network with an added *sharpness-aware* network (**SAGAN**), proposed by *Yi et al.* [19]. Figure 21 shows the overall architecture comprising three networks: the generator G, discriminator D and an auxiliary sharpness detector S, introduced for recovering sharpness in low contrast regions of noisy images. Here, the generator adopts the U-net [58] structure with 8 deep convolutional layers as well as long skip connections (see Figure 22). On top of the output being fed from one layer to the next, the long skip connections enable transmission to further selected layers, which allows for training a deeper network by "skipping" some of the layers. The discriminator architecture borrows from the pix2pix [62] GAN-base image-to-image translation framework and differentiates between overlapping image patches, as opposed to full images.

Equations (19) describe the respective adversarial $\mathcal{L}_{adv}(D,G)$ and pixel-wise $\mathcal{L}_{L_1}(G)$, $\mathcal{L}_{sharp}(G)$ loss functions adopted for training SAGAN. The authors chose to perform the min/max optimisations of G and D respectively in the least square sense rather than log-probability (Equation (17)). The mean square error and sharpness map differences are also backpropagated to update the generator weights. This results in a cumulative loss $\mathcal{L}_{SAGAN} = \arg\min_G\max_D(\mathcal{L}_{adv}(D,G) + \lambda_1\mathcal{L}_{L_1}(G) + \lambda_2\mathcal{L}_{sharp}(G))$, where λ_1 and λ_2 represent the weighting terms.

The sharpness detection network comprised a U-net architecture [58] and was trained on the set of 704 defocused images from [64]. The output sharpness map was created by the means of an analytic sharpness metric, proposed by *Yi et al.* (2016) [65].

$$\begin{aligned}
\mathcal{L}_{adv}(D,G) &= \mathbb{E}_{x,y\sim p_{data(x,y)}}[(D(x,y)-1)]^2 + \mathbb{E}_{x\sim p_{data(x)}}[D(x,\hat{y})^2],\\
\mathcal{L}_{L_1}(G) &= \mathbb{E}_{x,y\sim p_{data(x,y)}}[||y-\hat{y}||_{L_1}],\\
\mathcal{L}_{sharp}(G) &= \mathbb{E}_{x,y\sim p_{data(x,y)}}[||S(\hat{y})-S(y)||_{L_2}]. \qquad (19)
\end{aligned}$$

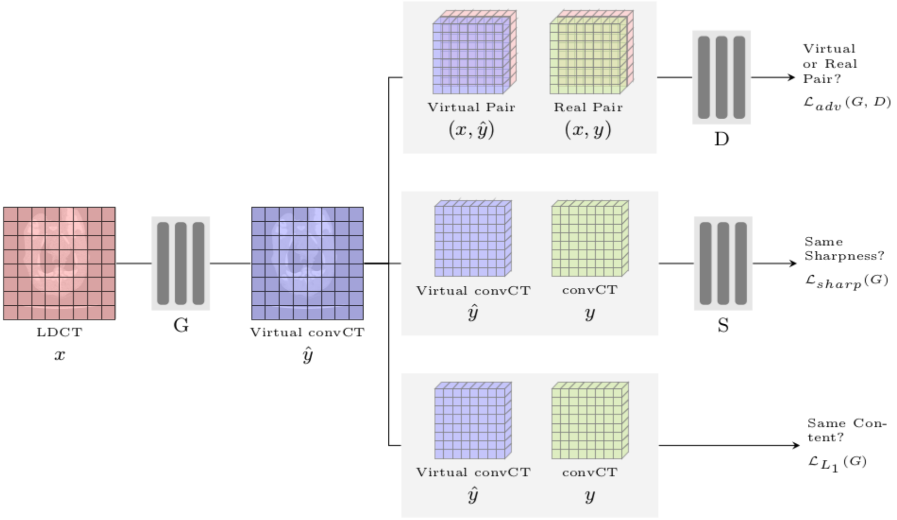


Figure 21: SAGAN components. G performs mapping $G : x \mapsto \hat{y}$ from input LDCT (x) to a *virtual* denoised CT $(\hat{y})$ such that $\hat{y}$ is as close to the real full dose target (y) as possible and D is not able to differentiate between y and $\hat{y}$. x paired with both y and $\hat{y}$ is also fed to the discriminator as auxiliary information for detecting the mismatch between image pairs. Network S compares the sharpness maps of the output of G with its respective ground truth. Image courtesy of *Yi et al.* [19].

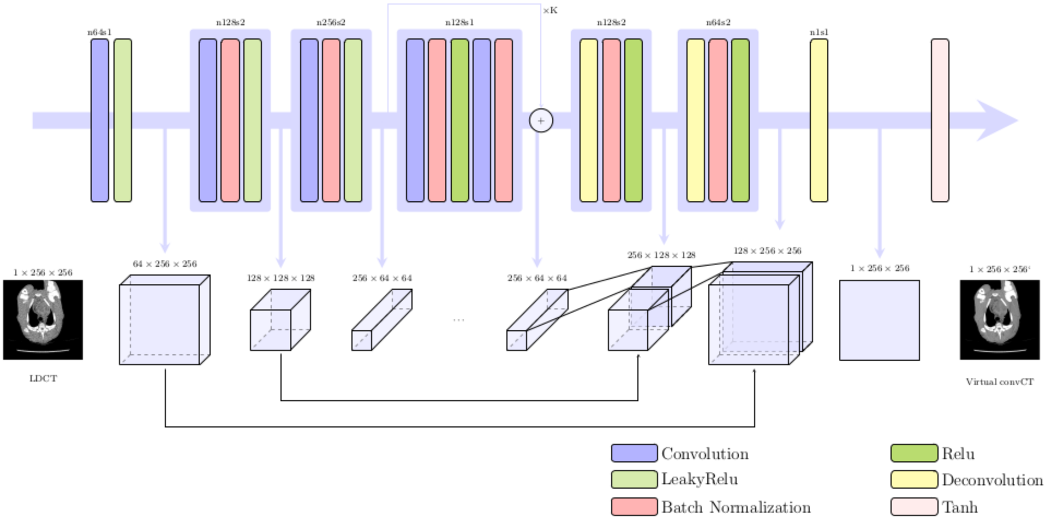


Figure 22: A schematics of the SAGAN generator. The network implements U-net structure [58] for feature learning and skip connections (represented by the black lines in the diagram) for efficiency. Coloured rectangles indicate the steps and constituents of each convolutional layer, while bottom row boxes show respective kernel sizes. Image courtesy of *Yi et al.* [19].

2.7.2 WGAN-VGG

A number of works have addressed the edge oversmoothing and fine detail loss associated with the per-pixel/voxel Mean Square Error (MSE) loss ([6], [20], [23]). To tackle these challenges, *Yang et al.* [6] put forward a GAN with Wasserstein Distance and Perceptual Loss (WGAN-VGG). There, the classic GAN loss function, described in Equation (17), was modified as follows:

$$\underset{G}{\min}\underset{D}{\max} L_{WGAN} = -\mathbb{E}_x[(D(x)] + \mathbb{E}_z[D(G(z))] + \lambda \mathbb{E}_{\hat{x}}[(||\nabla_{\hat{x}} D(\hat{x})||_2 - 1)^2], \tag{20}$$

where x and $\hat{x}$ are respective true and generated sample pairs and λ is a constant weighting parameter. The *Wasserstein distance*, represented by the first two terms, is a metric for calculating the distance between true and generated probability distributions. This is a direct difference measure as opposed to a regular GAN approach that tries to minimise that distance indirectly through the log-probability. Essentially, the former method compares the cost of generator mapping from one distribution to another directly, while the latter compares probability distributions of real and generated images. The last term in Equation (20) is an added *gradient penalty* for faster loss convergence. [6], [66] It has been demonstrated that using Wasserstein GAN rather than a standard GAN can result in a more stable optimisation process. [67]

An additional *perceptual loss* was also incorporated for retaining finer details, quantified as follows:

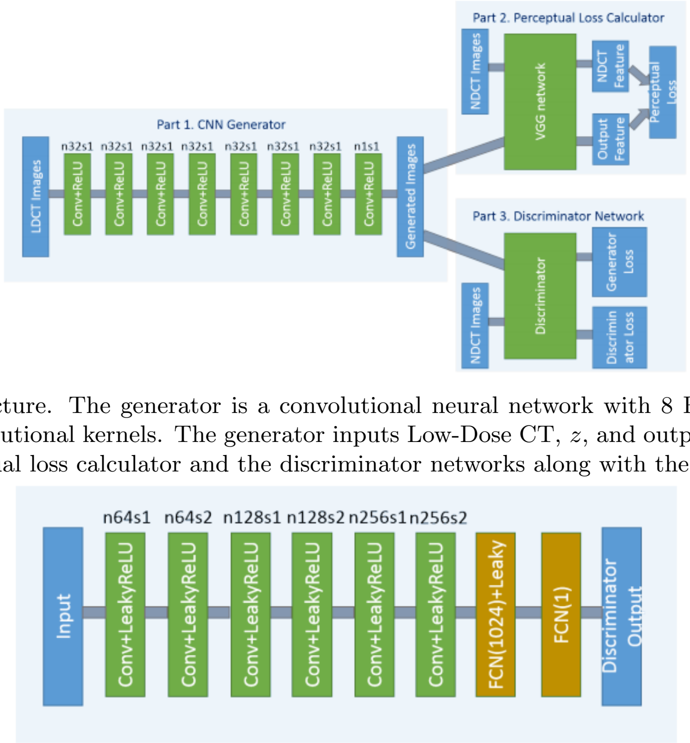
$$L_{VGG}(G) = \mathbb{E}_{(x,z)}[\frac{1}{whd}||\phi(G(z)) - \phi(x))||_F^2], \tag{21}$$

where the pre-trained VGG-19 [68] network (a deep CNN model for classifying image features) acts as a *feature extractor* ϕ, with the feature being the output of its last convolutional layer. Thus, perceptual loss computes the mean square difference between images in the established feature space instead of directly per-pixel/voxel. w, h and d are then the width, height and depth of the feature space respectively and $||\mathbf{A}||_F = \sqrt{\sum_{i=1}^m \sum_{j=1}^n |a_{i,j}|^2}$ for an $m \times n$ matrix $\mathbf{A}$. The overall cumulative loss of WGAN-VGG is then:

$$\min_G \max_D L_{WGAN} + \lambda_1 L_{VGG}, \tag{22}$$

where λ_1 is the VGG network weighting parameter.

Figure 23 demonstrates the WGAN-VGG architecture. The generator and discriminator comprise a CNN with 8 and 6 convolutional layers respectively and the pre-trained VGG serves as a perceptual loss calculator.



(a) The WGAN-VGG structure. The generator is a convolutional neural network with 8 ReLU-activated layers, each consisting of 32 3×3 convolutional kernels. The generator inputs Low-Dose CT, z, and outputs a denoised image, $G(z)$, which is fed to the perceptual loss calculator and the discriminator networks along with the ground truth image x.

(b) The WGAN-VGG discriminator network. It consists of 6 convolutional layers with consecutive 64, 128 and 256 filters of 3×3 kernel sizes. The discriminator is finalised with two fully-connected layers, where the penultimate one has 1024 outputs and the last layer downsamples to a single output.

Figure 23: A diagram of the WGAN-VGG architecture. Images courtesy of *Yang et al.* [6].

3 Methods

3.1 Methodological Approach

In the scope of this research the WGAN-VGG network [6] was re-trained on the local dataset to adjust the model hyperparameters for best results. A learning rate scheduler and early stopping were introduced to regularise the model performance. The cross-validation method was employed to increase the number of available test images. These were then denoised by the corresponding WGAN-VGG models, trained with the set of hyperparameters, inferred earlier.

To observe the effect of the dose level on the model output and ensure that the algorithm did not impede the quality of the images that were already low in noise, a range of simulated noise was applied to diagnostic dose images from the test set. Finally, resulting WGAN-VGG model was compared to a pre-trained SAGAN model [19] by first assessing both network performances on the data used for training SAGAN, followed by the local validation set.

3.2 Data

Data from two different studies were used in this project. In both cases routine dose FBP images were employed as the reference standard.

3.2.1 Clinical Data

The data were obtained from the ongoing study investigating dopamine transportation with the PE2I radiopharmaceutical in patients with suspected or diagnosed parkinsonism. [69], [70] Low dose CT scans are routinely performed alongside PET for attenuation correction in PET/CT. [71] High dose CTs were requested for patients with observed pathologies. The brain scans were obtained at *Rigshospitalet Glostrup, Copenhagen University Hospital* utilising a SIEMENS scanner (SOMATOM Definition AS) with a tube voltage of 120 kVp. Spatially-aligned diagnostic and low dose CTs were obtained with tube currents of 193 mAs (CTDI$_{vol}$ = 59.99 mGy) and 60 mAs (CTDI$_{vol}$ = 6.83 mGy) respectively (an equivalent of approximately 1.5 mSv and 0.2 mSv, corresponding 90% reduction in dose in LDCT as compared to diagnostic). All images were FBP-reconstructed to 111 × 2 mm slices with the H19 convolution kernel (dimensionality of 512×512 pixels with 0.59×0.59 mm^2 spacing). The patient data were appropriately anonymised in accordance with relevant data protection standards. A total of 27 different patient datasets were utilised over the course of the project (9 of which were female and 18 male, with a mean age of 71.26 ± 1.64).

3.2.2 Piglet

In this case, the data consisted of full-body CT scans of a single deceased piglet at dose levels ranging from full dose to reductions of 50%, 25%, 10% and 5% (tube potential was set to 100 kVp for all series) and various reconstruction methods respectively. Filtered backprojection alone was used in the scope of this project. 906 image slices of 0.625 mm thickness and 0.41×0.41 mm^2 pixel spacing (512×512 matrix) from the full dose (tube current 300 mAs, CTDI$_{vol}$ = 30.83 mGy with effective dose of 14.14 mSv) and the 95%-reduced scans (tube current 15 mAs, CTDI$_{vol}$ = 1.54 mGy with effective dose of 0.71 mSv) were selected for the comparative network testing experiment. The dataset was previously acquired for SAGAN training by *Yi et al.* [19] and downloaded with the author's consent. This data was used exclusively for comparing SAGAN [19] performance with that of WGAN-VGG [6].

3.2.3 Training, Testing and Validation Data Split

The size of the patient cohort used for selected experiments was defined by the number of paired data made available at the relevant time. The first model was trained on 10 patients. Most of the models trained for the purpose of identifying optimal network hyperparameters had a training set of 15 clinical patients, while the model employed for the learning rate scheduler and simulated noise experiments comprised a total of 22 training patients. Two separate patients were held out for model validations on all occasions.

Due to a limited size of the data sample, the ability of the best performing model to generalise to unseen data was tested by applying the cross-validation method. [73] The two patients used as a validation set during the fine-tuning stage were excluded from this statistical experiment to reduce bias in the evaluation of model fits. The remaining 25 datasets were split into 5 subsets (*folds*) by isolating 5 patients and training on the rest of the data. The resulting five models, each trained on 20 patients, were then evaluated with respective retained testing subsets. This way, the entire cohort participated in both validation and training, with each example serving as validation exactly once, thus, producing 25 unique testing outputs and performance results averaged over five models. Figure 24 shows the data split distribution for the clinical dataset.

For the evaluation of the 5-fold outcome images, randomly selected patient slices from the middle of the brain regions were selected from all 5-fold model test sets. This was done with the aim of each example to serve as a sensible respective group representative, while also omitting the slices mostly depicting air and possible spatial misalignments across the entire image series. The final quantitative evaluation was performed on 5 different image triplets (low-dose, output, diagnostic), where each output was denoised with a separate model and each image set was a subset of respective model training data.

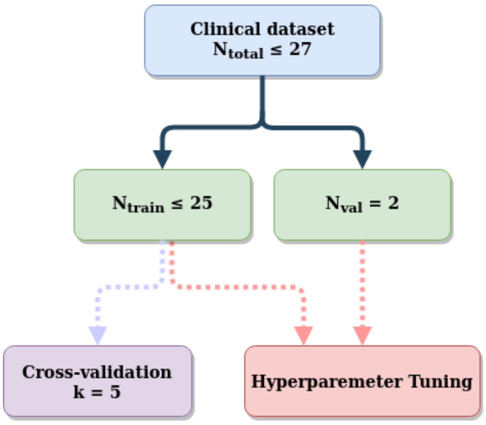


Figure 24: A schematic of the clinical data split for the purposes of different experiments. The two validation patients utilised for hyperparameter tuning were excluded and the inferred parameters were used to train the models at the cross-validation stage.

3.2.4 Simulated Dataset

A simulated noise dataset was utilised to quantify the relation between the initial noise level and the magnitude of the denoising in the output images. To compare the denoising abilities across the five folds, a controlled range of noise was applied to diagnostic dose images from the five selected sets, described above.

`Matlab` was employed to simulate the effect of dose on the reconstructed image quality (courtesy of *Yi et al.* [74]). Fan beam geometry was used to transform the clinical data into the sinogram domain. The noise levels were controlled by varying the number of incident flux photons, following Equation (4), for $N_0 = 1 \times 10^6, 3 \times 10^6, 5 \times 10^6, 1 \times 10^7$ (mimicking the range utilised in [19]). Electrical noise was discarded for the purposes of this experiment. Each image was then denoised with the appropriate corresponding model and the PSNR with SSIM compared across folds.

3.3 WGAN-VGG Implementation

The WGAN-VGG (refer to Section 2.7.2 for details of the architecture) training/testing model by *Yang et al.* [72] was implemented with `TensorFlow` 1.14.0 (Python v3.6) on NVIDIA Titan GPU. Visual analysis and model fits were performed using the `TensorBoard` toolkit.

3.3.1 Network Training

All the models were optimised using the *Adam* algorithm [56] with all the parameters, apart from the learning rate α and VGG-19 network weight λ_1, intact ($\beta_1 = 0.5$, $\beta_2 = 0.9$, $\lambda = 10$) as prescribed by the authors [6]. The various model generators G were trained for a range of iterations in mini-batches of 128.

3.3.2 Data Pre-/Post-Processing

All the input DICOMs files were resampled into 512×512 32-bit images slice-by-slice. The minimal and maximal pixel values were mapped to -1024 and 3075 respectively to cover the clinical HU range. The images were then divided into 64×64 patches and respective low-/diagnostic dose pairs concatenated for training. The greyscale CT images were duplicated to make RGB channels before being fed to the pre-trained VGG network. The denoising model was applied on whole-size test images. The resulting pixel data from the network output was then written to corresponding DICOM files as 16-bit integers.

Training Iterations

The optimal number of iterations was investigated with the WGAN-VGG model trained on 10 clinical patients. The model was validated on two separate patients after 200, 600 and 850 thousand iterations and respective outputs compared. The effect of continuing model training after initial convergence was also observed by training a model on 2 patients, followed by the addition of 8 extra patients with persisting iterations.

3.3.3 Hyperparameter Tuning

As the VGG perceptual loss effectively accounts for the structural difference between the input and generated images and is intrinsically linked to the accuracy of model fit to both noise distributions, the respective loss function and reciprocal fit qualities were used to evaluate model performance. The perceptual loss curves and PSNR between LDCT input and denoised output, according to Equations (21) and (24) respectively, were used to estimate the effects of the initial learning rate α and the weight of the VGG network λ_1 on the WGAN-VGG model convergence. The specific choice of these parameters was justified by two reasons. Firstly, even though the learning rate is among one of the most important parameters, it cannot be calculated analytically for tailored cases. [52] Secondly, it was compelling to investigate the possible refinements in denoising quality introduced by the proposed perceptual similarity comparator. The models were trained with the dataset of 15 patients for 120 thousand iterations and validated on 2 separate cases.

The potential benefits of introducing the VGG network as a supplement to the GAN model were assessed by adjusting the weighting parameter λ_1, responsible for the trade off between the perceptual and adversarial losses, and comparing the results with the proposed value of $\lambda_1 = 0.1$. One of the models was also trained with Wasserstein GAN alone (by setting λ_1 to 0). The initial learning rate was kept at the original value of $\alpha =$ 1e-5. [6]

Model Regularisation

To further solidify the choice of an appropriate value of α for the optimal model convergence, a learning rate scheduler was introduced to the Adam optimiser during the hyperparameter tuning process. The initial learning rate α_0 was gradually reduced with an exponential time decay as follows:

$$LR_{decay} = \alpha_0 \times 0.95^{\frac{step}{10000}}, \tag{23}$$

where α_0 was set to decay by 5% every 10000^{th} iteration step. Two models with different initial learning rates (α_0 = 1e-5 and 5e-6; λ_1=0.15) were trained with 22 patients for 120 thousand iterations in order to investigate whether gradually reducing the α_0 value after several training epochs would have ultimately resulted in better fit parameters later on. A third model with the slowest learning rate α = 1e-6 (λ_1=0.15) was trained without decay scheduling for comparison.

Lastly, the resulting model trained on 22 patients with the best performing set of hyperparameters (identified by the performance on the validation set of 2 patients) was trained for 800 thousand iterations. The model performance was observed after 100, 200, 600 and 800 thousand iterations to evaluate how fit accuracy scales with the number of training iterations.

Early stopping (without the learning rate scheduler) was implemented for the final models trained for the 5-fold cross-validation (α = 1e-6, λ_1=0.15). This way, appropriate numbers of training iterations for respective model convergence were ensured by prescribing training halt once the perceptual loss stopped decreasing or had plateaued (showed no improvements) for at least 200 thousand iterations. All five resulting models demonstrated optimal convergent behaviour around 650 thousand iterations. Both the learning rate scheduler and the early stopping were implemented with the `Keras callback API`.

3.4 SAGAN Implementation

The pre-trained SAGAN [19] model was obtained from author's GitHub repository [74]. A more detailed description of the network can also be found in Section 2.7.1 of the background theory. CUDA version 9.1 with cudNN 9.0 library and Torch framework (Python v3.7) were employed to run SAGAN. The testing is performed on full images of 512×512 resolution. This network was not re-trained on the local data and was included in the investigation solely for the purpose of comparing performance between the two different architectures.

3.5 Network Comparison

To investigate the feasibility of SAGAN implementation and compare the denoising abiliies of different network architectures, the pre-trained SAGAN model was validated on the training data as well as tested against the previously unseen local clinical data (the validation set of 2 patients described above). Both datasets were also denoised with the WGAN-VGG model ($\alpha = 1 \times 10^{-6}$, $\lambda_1 = 0.15$, trained on clinical 22 patients for 650 thousand iterations).

3.6 Evaluation Metrics

To get an objective estimate of the denoising power of the model, the noise levels in the output images were evaluated quantitatively in terms of their PSNR, SSIM as well as mean CT numbers and standard deviations of selected uniform regions (similarly to [6], [19], [20]). Finally, the quality of the noise suppression was scrutinised by the on-site radiologist.

PSNR

One quantitative measure for spatially-aligned CT images is the *peak signal-to-noise ratio* (PSNR), defined as follows:

$$PSNR = 20 \log_{10} \frac{255}{\sqrt{MSE}}, \tag{24}$$

where 255 is maximal pixel value for an 8-bit image and $\sqrt{MSE}$ is the root mean square error between the testing and ground truth images. [1] All the analysed 16- and 32-bit images were linearly scaled to fit the 0-255 range.

SSIM

The *Structural Similarity Index* (or SSIM) is an popular measure of perceptual difference between two images alternative to PSNR. The value is obtained by comparing image textures (specifically luminance, contrast and structure) as opposed to error differences (see [75] for the introduction of the concept). The SSIM of the denoised images with the respective ground truths were calculated with the Python skimage library.

ROI STD

For this evaluation, mean CT numbers (in Hounsfield Units) along standard deviations were obtained for similar uniform regions of same radius across all inspected images. All regions of interest (ROI) were limited to white matter in the frontal lobe to minimise the variation in CT numbers (and noise) due to anatomy. Values from ROIs in the spatially-aligned series were obtained via the minc toolkit. [76]

Clinical Evaluation

A blinded test of the denoising method effectiveness was performed by a local radiologist (an MD with over 10 years of experience in diagnostic radiology). All of the denoised CT images (25 × 111 slices) generated by the network along with their respective diagnostic and input low doses were randomised. The resulting 75 patients were subjectively scored, one after another, based on the overall image quality (1 = "poor quality, unusable for clinical purpuses"; 2 = "average quality, but still usable"; 3 = "good quality, perfectly usable"). The significance of the outcome was assessed with a paired t-test with the upper-tailed alternative hypothesis assuming that mean output score was greater than that of the input.

4 Results

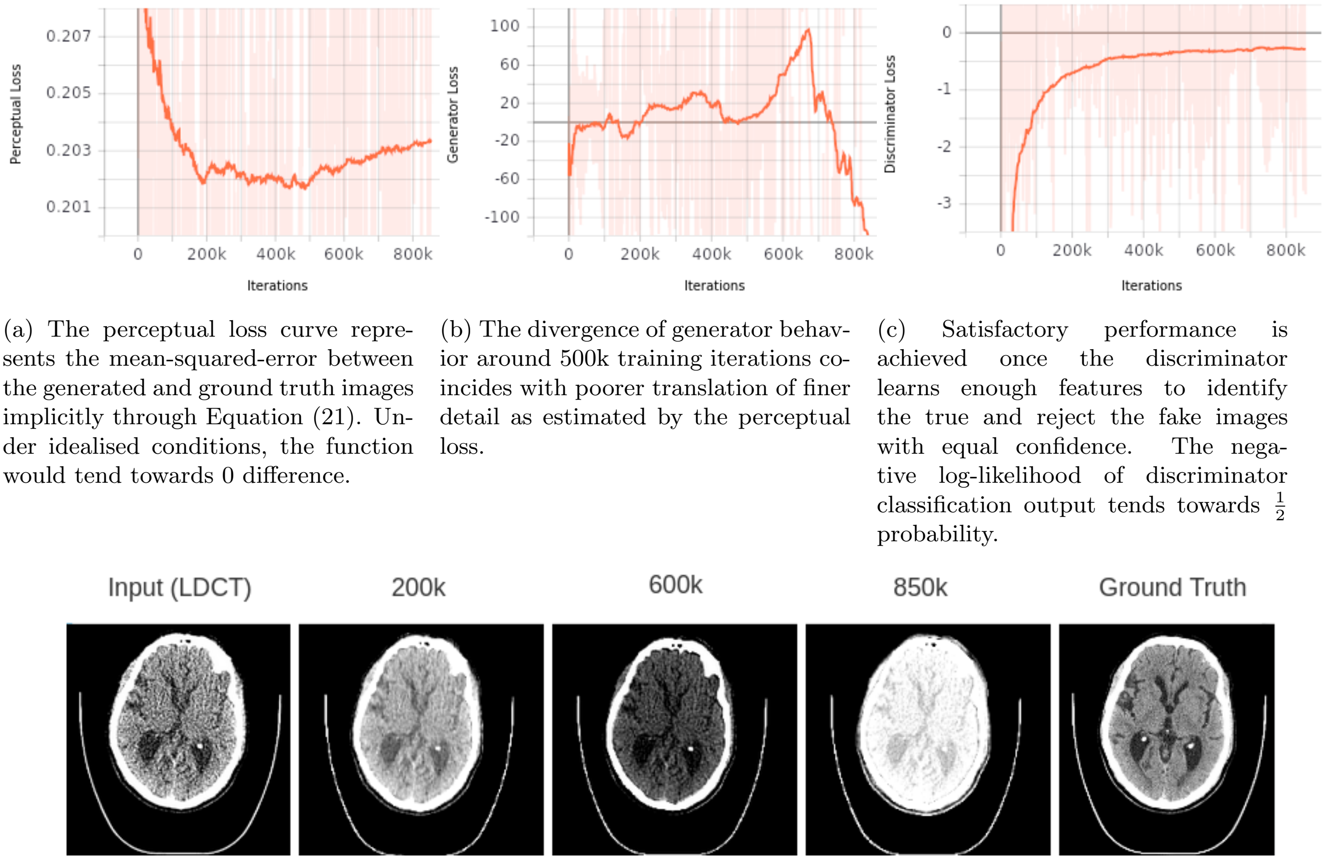
4.1 WGAN-VGG Performance Optimisation

The following section considers various approaches to WGAN-VGG [6] (Section 2.7.2) network training optimisation and model regularisation. The optimal set of parameters for desirable model convergence was inferred empirically through monitoring performance during training and comparing respective output image qualities from a range of trained models. The ability of the best performing model to generalise to unseen data was then validated through using the 5-fold cross-validation method.

4.1.1 Training Iterations

Figure 25 is the denoising result of WGAN-VGG output trained on 10 different clinical patients. The juxtaposition of the generator and discriminator loss curves (Figures 25b - 25c) suggests that the network stabilises after approximately 200 thousand iterations and such trend continues up until around 500 thousand iterations where the generator loss starts to diverge. This is also reflected by the perceptual loss curve, which increases in numerical value as the generator quality degrades (Figure 25a).

Figure 26 also shows the deteriorating effects of increasing the training data size fed to a WGAN-VGG model with a previously optimised discriminator.



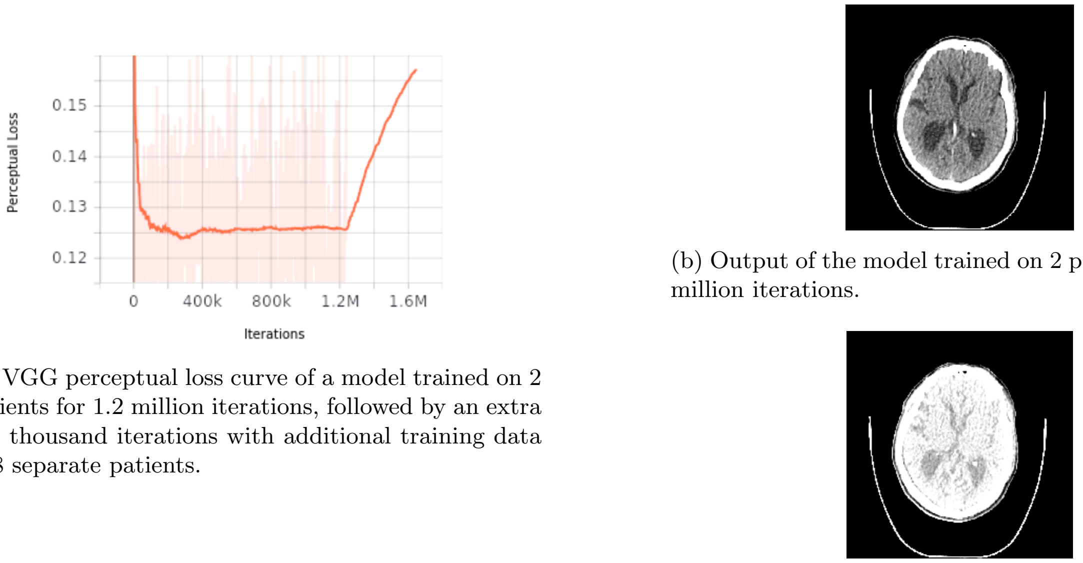
(a) The perceptual loss curve represents the mean-squared-error between the generated and ground truth images implicitly through Equation (21). Under idealised conditions, the function would tend towards 0 difference.

(b) The divergence of generator behavior around 500k training iterations coincides with poorer translation of finer detail as estimated by the perceptual loss.

(c) Satisfactory performance is achieved once the discriminator learns enough features to identify the true and reject the fake images with equal confidence. The negative log-likelihood of discriminator classification output tends towards $\frac{1}{2}$ probability.

(d) Comparison of model output after training for 200, 600 and 850 thousand iterations. Continuing to train the model past the discriminator convergence point eventually leads to contrast abnormalities and structural loss in generated images.

Figure 25: Visual denoising quality of WGAN-VGG trained on 10 patients for an increasing number of training iteration.



(a) VGG perceptual loss curve of a model trained on 2 patients for 1.2 million iterations, followed by an extra 400 thousand iterations with additional training data of 8 separate patients.

(b) Output of the model trained on 2 patients after 1.2 million iterations.

(c) Same model output following the intoduction of 8 more patients to the training set.

Figure 26: The effect of increasing patient cohort in a converged GAN-based model.

4.1.2 Hyperparameter Tuning

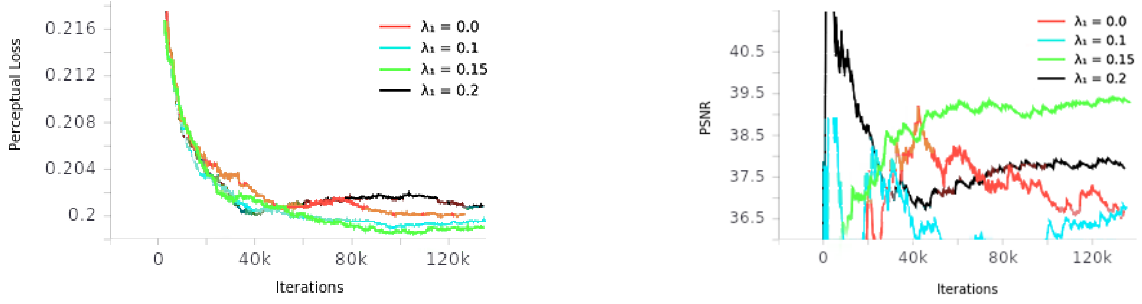
The following section presents various WGAN-VGG outputs from models trained with different sets of hyperparameter values as well as associated quantitative results.

VGG Weight, λ_1

The perceptual loss and output PSNR functions, plotted as a mean of 128 batches against respective iterations for models with varied VGG weights (λ_1 = 0.0, 0.1, 0.15, 0.2 and a fixed learning rate α = 1e-5) are shown in Figure 27. An example of denoised outputs from each models can be seen in Figure 28. The intermediate value of $\lambda_1 = 0.15$ produced the most plausible contrast in the output image (Figure 28). This result was also supported numerically in the form of the highest mean output PSNR, as compared to other values of λ_1, with an identical setup otherwise (Table 1 demonstrates mean batch PSNR values across trained models). Finally, $\lambda_1 = 0.15$ yielded the highest average PSNR and SSIM values (as estimated on the whole images from the validation set, shown in Table 2) among the four models in question.

Learning Rate, α

Along with the initial optimiser learning rate $\alpha = 1 \times 10^{-5}$ (1e-5) selected by the authors [6], [66], performances with $\alpha = 1 \times 10^{-4}$ (1e-4) and $\alpha = 1 \times 10^{-6}$ (1e-6) were evaluated. The results are shown in Figures 29 and 30 as well as Tables 1 and 2.



(a) Perceptual loss curves from models with $\lambda_1 = 0.0$, 0.1, 0.15, 0.2. All four models, including the one comprising GAN only, $\lambda_1 = 0.0$, demonstrated the ability to decrease the loss at a similar rate.

(b) The models, trained with $\lambda_1 = 0.0$, $\lambda_1 = 0.15$ and $\lambda_1 = 0.2$, all produced output PSNR values higher than the original parameter of $\lambda_1 = 0.1$ on average (see Table 1 for mean PSNR values). However, due to all three curves outlining the noise distribution in distinctly different manners, it is ambiguous to draw any meaningful conclusion based solely on the presented PSNR values.

Figure 27: Training models with different trade-offs between the perceptual and adversarial losses.

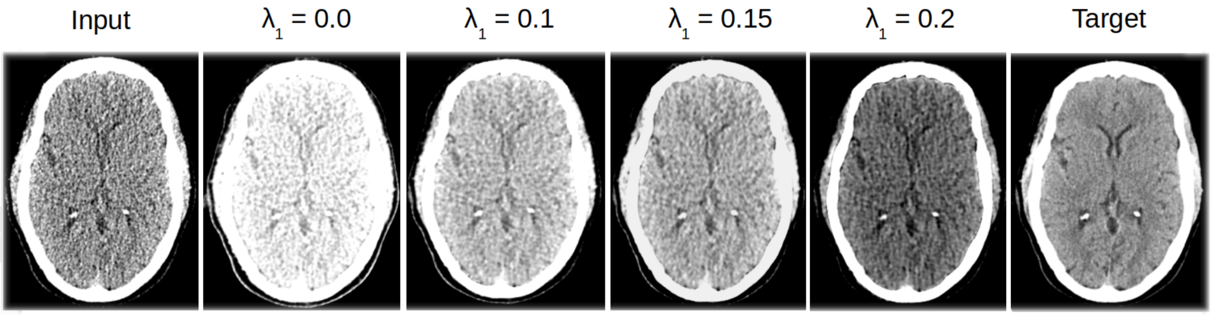
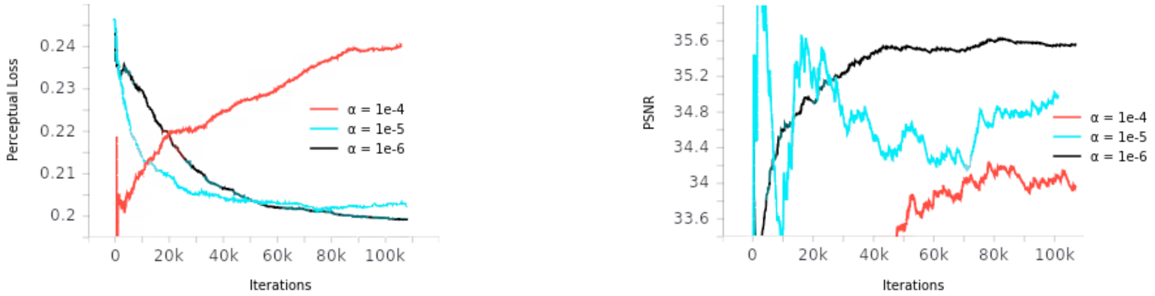


Figure 28: Visual comparison of model outputs trained with $\lambda_1 = 0.0, 0.1, 0.15, 0.2$. Where the image quality was concerned, the structural details appeared to translate better, as compared to the ground truth, for images produced with models where the VGG network was on. Slight increment in the λ_1 value yielded better contrast in the denoised image as compared to the ground truth.



(a) Perceptual loss curves produced with various initial learning rates. The slopes of the blue (α = 1e-5) and the black (α = 1e-6) curves indicate similar loss minimisation speed, while the red (α = 1e-6) model curve fails to decrease the perceptual loss all together.

(b) Estimated network output peak signal-to-noise evolution for models with different initial learning rates (see Table 1 for respective mean PSNR values).

Figure 29: Visual analysis of convergent behaviour of different models based on their initial learning rates. The slowest α = 1e-6 demonstrated the most stable behaviour throughout training process as well as achieving the lowest perceptual loss values out of the three variants, outperforming the initial value of α = 1e-5 starting from around 50 thousand iterations.

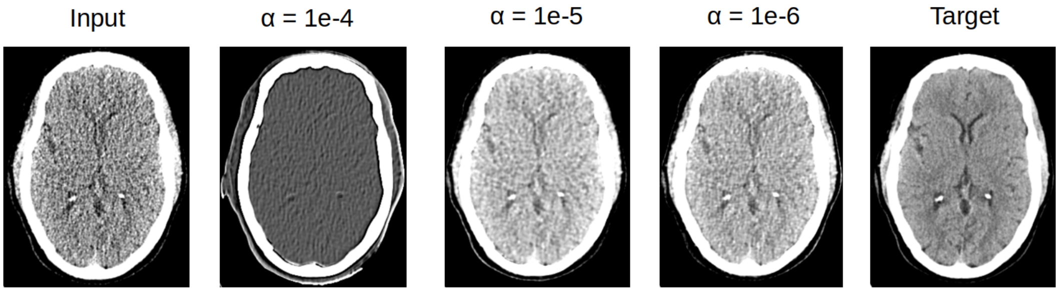


Figure 30: Adjusting initial learning rates for the WGAN-VGG model. Following the results shown in Figure 29, α = 1e-5 and α = 1e-6 yield similar visual quality, while the model with learning rate α = 1e-4 fails to converge, producing meaningless output.

Hyperparameters	Output PSNR
LDCT	34.5790 ± 0.3290
α = 1e-5, $\lambda_1 = 0.1$	34.8379 ± 0.3685
α = 1e-5, $\lambda_1 = 0.0$	38.7587 ± 0.2839
α = 1e-5, $\lambda_1 = 0.15$	38.8953 ± 0.3457
α = 1e-5, $\lambda_1 = 0.2$	37.6991 ± 0.2905
α = 1e-4, $\lambda_1 = 0.1$	32.7359 ± 0.2565
α = 1e-6, $\lambda_1 = 0.1$	35.5533 ± 0.3841
α = 1e-6 $\lambda_1 = 0.15$	39.1430 ± 0.1415

Table 1: Mean batch peak signal-to-noise estimated by the selected models during optimisation for the denoised patches in accordance with Equation (24), also plotted in Figures 27b and 29b against training iterations. All the models were trained on a dataset of 15 patients for the same number of iterations. LDCT value is shown for comparison. The best performance is highlighted in red.

Model	PSNR	SSIM
LDCT	86.0474 ± 0.8770	0.9766 ± 0.0014
α = 1e-5, $\lambda_1 = 0.1$	86.0557 ± 0.8900	0.9817 ± 0.0011
α = 1e-5, $\lambda_1 = 0.0$	86.8431 ± 0.9440	0.9826 ± 0.0011
α = 1e-5, $\lambda_1 = 0.15$	87.0580 ± 1.0824	0.9805 ± 0.0013
α = 1e-5, $\lambda_1 = 0.2$	86.3361 ± 0.9283	0.9812 ± 0.0012
α = 1e-4, $\lambda_1 = 0.1$	83.0017 ± 0.7874	0.9665 ± 0.0013
α = 1e-6, $\lambda_1 = 0.1$	87.8318 ± 1.1314	0.9846 ± 0.0011
α = 1e-6, $\lambda_1 = 0.15$	87.7527 ± 1.1762	0.9827 ± 0.0013

Table 2: Comparison of the quantitative validation outputs from models with various choices of hyperparameters, previously introduced in Table 1 and associated with Figures 28 and 30. The stated PSNR and SSIM values were obtained by averaging the results across 111 output image slices from the 2 validation patients.

***

For both the PSNR values, estimated per training epochs and from the denoised validation set images, the learning rate α = 1e-6 and VGG weight $\lambda_1 = 0.15$ produced the highest mean output (Tables 1 and 2). Thus, these parameters were combined to train a separate model for best numerical improvement. Despite the fact one model (same learning rate, $\lambda_1 = 0.1$) yielded higher PSNR and SSIM values in the validation images, the choice of the best performing model was also based on the visual appearance of the final output with respect to the ground truth.

Learning Rate Schedulers

Despite yielding the highest PSNR gain (Table 2), the slowest learning rate α = 1e-6 demonstrated some underfitting behaviour in the noise distribution as compared to the other models (Figure 29b).

For this experiment two models with learning rate schedulers, as prescribed by Equation (23), and different initial learning rates (α_0 = 1e-5 and α_0 = 0.5e-6), as well as one model with α = 1e-6 and no scheduler were trained on 22 patients for 120 thousand iterations. The results can be seen in Figure 32 and Table 3. The model with initial learning rate α_0 = 1e-4 was excluded from the analysis due to its failure to converge. Despite being less successful at fitting the true noise distribution right away, the slowest learning rate, without the addition of a time decay, achieved the highest estimated PSNR gain, which was supported by the mean PSNR and SSIM of the test images, previously unseen by the trained models (Table 3).

Model		PSNR	SSIM
	LR Scheduler		
α_0 = 1e-5	Yes	86.3479 ± 0.9325	0.9812 ± 0.0012
α_0 = 0.5e-6	Yes	86.3686 ± 0.9624	0.9814 ± 0.0012
α = 1e-6	No	87.4016 ± 1.1056	0.9835 ± 0.0012
+200k iterations		87.5785 ± 1.1242	0.9831 ± 0.0012

Table 3: The effect of implementing a learning rate scheduler on the statistical properties of the denoised images. All models were trained on 22 patients and validated on the 2 separate patients. The model output with no learning rate scheduler was repeatedly tested after training the model for additional 200 thousand iterations.

4.1.3 WGAN-VGG Model Regularisation

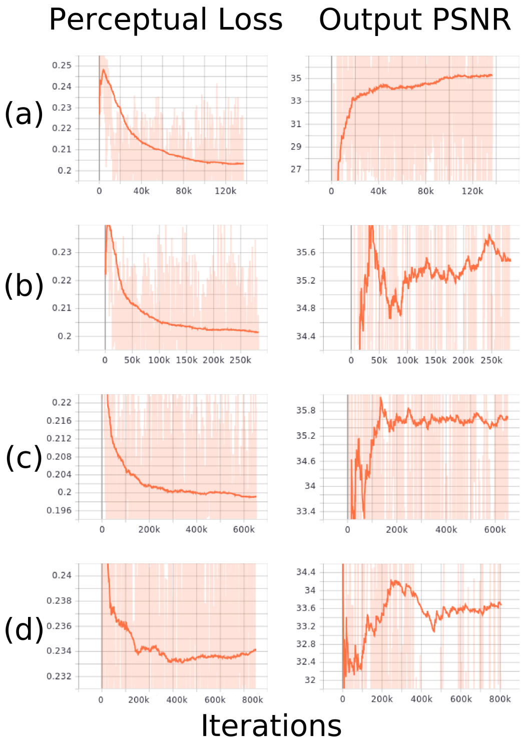
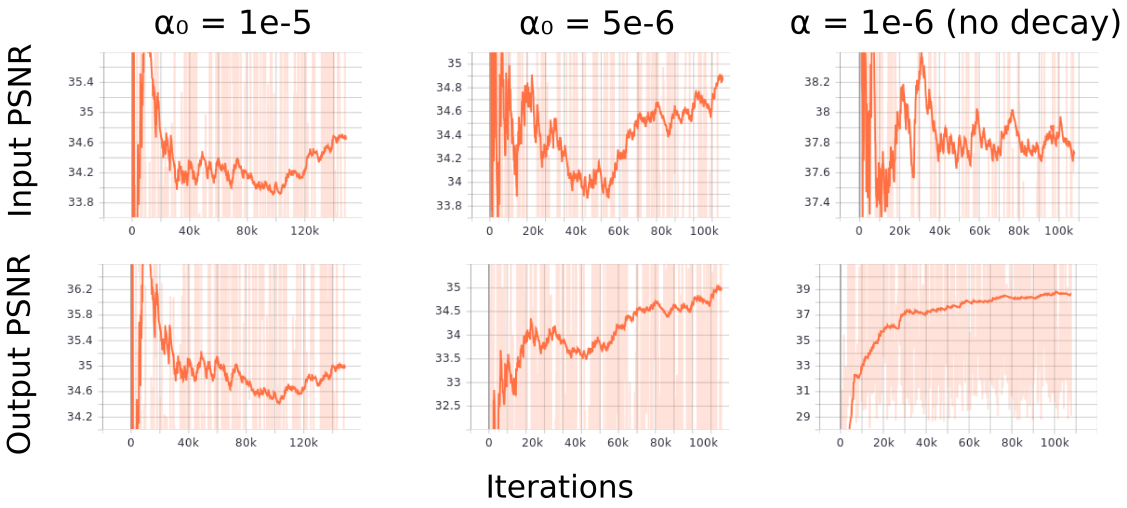
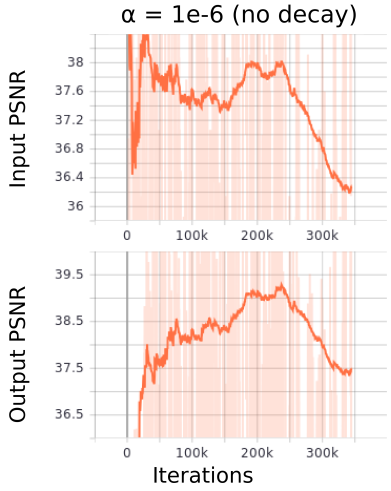


Figure 31: Perceptual loss and associated PSNR estimations for input and output images as functions of training iterations for the same model after (a) 120, (b) 250, (c) 600 and (d) 800 thousand iterations. The presented model was trained with 22 patients (α = 1e-6, λ_1 = 0.15) and validated on 2.

In addition to selecting an optimal set of hyperparameters that would best generalise to the validation data, it was important to identify the appropriate number of iterations prior to performing cross-validation, since it proved to scale with the size of the training cohort (Figures 26 and 32). Consistent with the results, shown in Figures 29b and 32b, the larger number of patients required longer training time, underfitting when the iterations were too low and overfitting when too high. The perceptual loss appears to plateau after 250 thousand iterations. However, some further training was required for the model to adjust the fit parameters accordingly and achieve a stable PSNR increase, as seen by comparing (b) and (c) in Figure 31. Naturally, persisting iterations had caused a significant overfit and misrepresentation of all three functions and, hence, discrepancy in true value estimations, as compared to other models.



(a) The input and output image PSNR distributions as estimated by the models during respective network optimisations. The faster the learning rate, the better the noise representation appeared to be early on with α_0 = 1e-5 bearing the most resemblance to the input and α_0 = 1e-6 completely underfitting the data. The mean output PSNR values comprised 34.7124 ± 0.2985 for α_0 = 1e-5, 35.2207 ± 0.2419 for α_0 = 5e-6 and 38.3419 ± 0.3689 for α = 1e-6.



(b) Training the model with the slowest learning rate for longer allows it to capture the aspects of the noise distribution better. The reciprocated mean PSNR increased to 39.4039 ± 0.3154 .

Figure 32: The effects of implementing a learning rate scheduler in conjunction with different initial learning rates α_0 on models trained with 22 patients for 120 thousand iterations. (a) Left to right: α_0 = 1e-5 with LR scheduler, α_0 = 0.5e-6 with LR scheduler, α = 1e-6 and no scheduler. (b) Further iterations of the model with no LR scheduler for additional 200 epochs. Even though it took longer for the model with slower learning rate to learn the data distribution, it produced the highest PSNR gain. Moreover, there was a further increase in its value (by 1.062), associated with subsequent iterations. The quantitative results on the final output images of the models in question can be found in Table 3.

4.2 Denoising Quality Evaluation

The statistical and aesthetic attributes of the final output images from the 25 patients, tested with the associated 5 cross-validation models (labelled 'FOLDN', where N was the consecutive training number), were scrutinised for potential upscale of low dose image quality.

The five model outputs along with their input low doses and the target diagnostic images are shown in Figure 34. The respective image PSNR and SSIM values, along with LDCT values for comparison, are stated in Table 4. The mean PSNR and SSIM values across the image series for all models and their test data can be found in Appendix A.

Additionally, the mean CT numbers and standard deviations of similar circular regions of the same radii were compared for the input-target-output sets across all models (Figure 33 and Table 5).

An increase was observed in the PSNR and SSIM values across all models and associated validations, excluding one patient in FOLD4 (found in Appendix A).

The WGAN-VGG output image mean CT values of the ROI in Figure 33 were smaller on three occasions and higher on two, as compared to both the low and target dose images. Nevertheless, denoised output yielded much smaller STD values, akin to those of the diagnostic dose images.

	FOLD1		FOLD2		FOLD3		FOLD4		FOLD5	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
LDCT	84.2777	0.9791	83.0414	0.9743	79.1954	0.9600	78.2294	0.9515	88.2097	0.9859
Denoised	85.2071	0.9881	84.4909	0.9865	80.1083	0.9715	78.4141	0.9623	88.4921	0.9925

Table 4: PSNR and SSIM values associated with images shown in Figure 34. The PSNR and SSIM values increased by 0.7518 ± 0.2330 and 0.0100 ± 0.0010 on average across models for this sample in agreement with values in Appendix A.

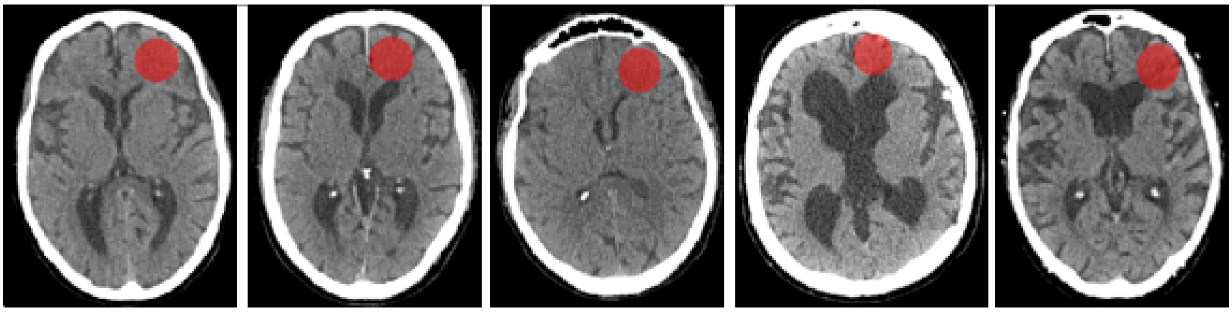


Figure 33: Examples of selected regions of interest (ROI) from individual high dose scans (associated with image sets shown in Figure 34) used for evaluating mean CT numbers across test images. The radii of all five circular selections were 15 mm.

	FOLD1		FOLD2		FOLD3		FOLD4		FOLD5	
	Mean	STD	Mean	STD	Mean	STD	Mean	STD	Mean	STD
LDCT	37.3201	10.9904	37.9746	14.7385	38.7308	11.3307	43.0408	17.3987	35.0602	13.5254
WGAN-VGG	34.7545	6.6432	34.1053	8.8168	34.0956	6.6103	45.1605	11.4934	37.7952	8.4015
Diagnostic	36.9925	5.2524	38.8693	6.6957	38.5066	5.2611	41.8147	12.0406	35.2038	8.2942

Table 5: Mean HU values of the regions of interest (indicated in Figure 33) and their standard deviation.

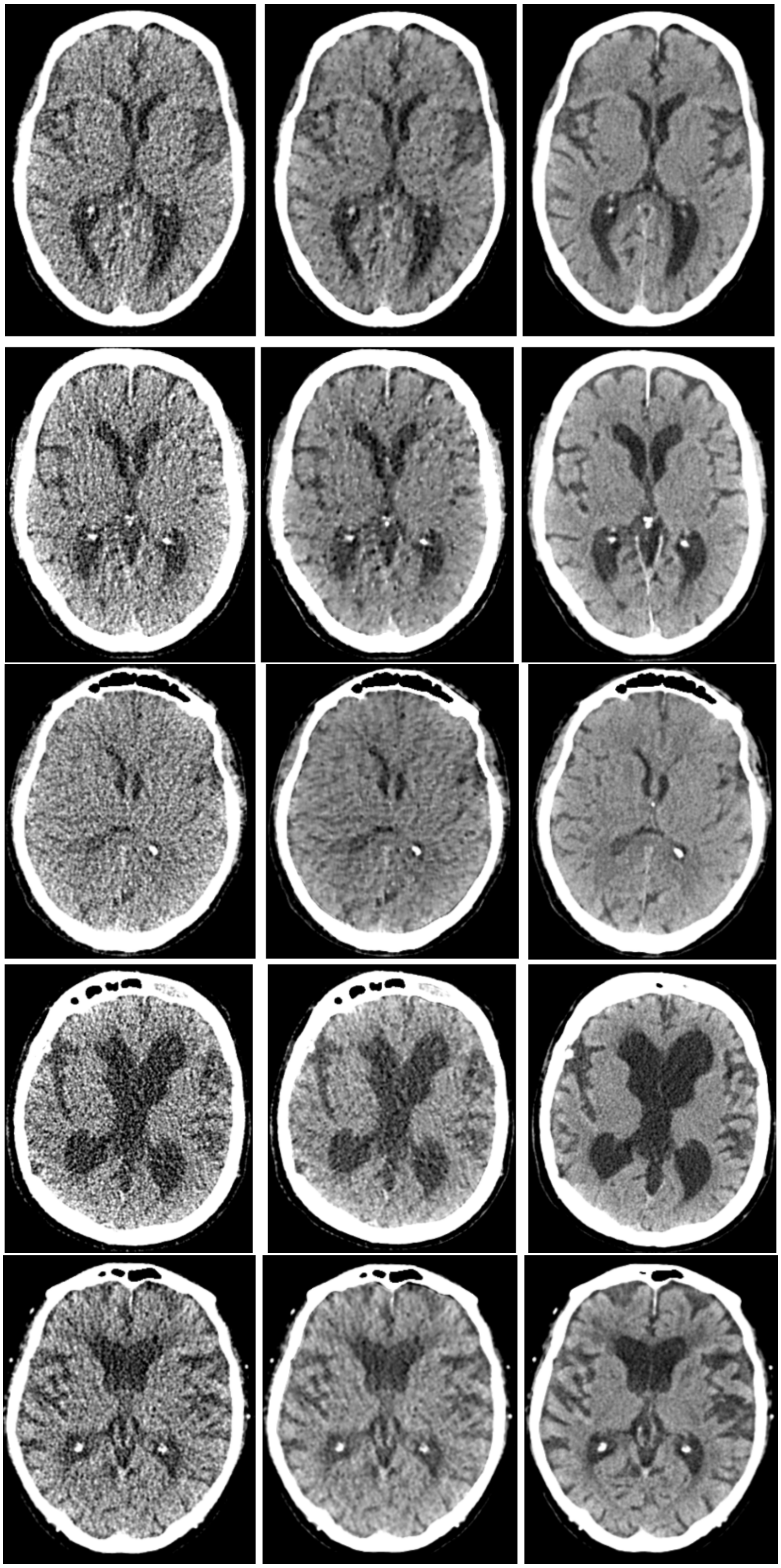


Figure 34: Final model denoising results from the 5 statistical folds (left to right): LDCT, denoised output, diagnostic. Top to bottom: FOLD1, FOLD2, FOLD3, FOLD4, FOLD5.

4.3 Clinical Evaluation

Table 6 shows the mean quality rating for the randomised blinded scoring of the low dose, denoised output and diagnostic dose images from the 25 clinical patients, according to procedure described in Section 3.6. The diagnostic dose scans were awarded a perfect rating on all occasions by the radiologist. The final model denoised output yielded a 0.16 improvement from LDCT (p = 0.08), with only 2 cases rated worse quality post-denoising.

Even though there appeared to be an improvement in image quality, some of the denoised images were reported to be "slightly different" in perceived structure as compared to other images in the evaluation set. Overall, the denoised images were deemed usable for evaluation and distinguishing smaller structures, like the ventricles. However, the clinical expert concluded that the resulting images were still insufficient for carrying out a confident diagnosis.

	LDCT	WGAN-VGG Output	Diagnostic
Quality Score	1.36 ± 0.12	1.52 ± 0.12	3.0

Table 6: Quality rating of the 25 clinical test patients (mean ± std) based on the 3-point scale. The paired t-test of LDCT and WGAN-VGG output resulted in a $t_{score} = 1.14$ and the one-tailed p-value of $p = 0.08$.

4.4 Simulated Noise Experiment

Figure 35 is an example of the effect of varied simulated noise levels on a diagnostic dose image and respective denoised outputs. As expected, the performance improves when the noise level is lower. All models demonstrated comparable results, with output scaling linearly with given dose levels (Table 7). Two out of five models followed this trend, but did not achieve an increase in PSNR , despite visual similarity of the resulting output (FOLD4 and 5). The improvement in SSMI values post-denoising was observed on all occasions.

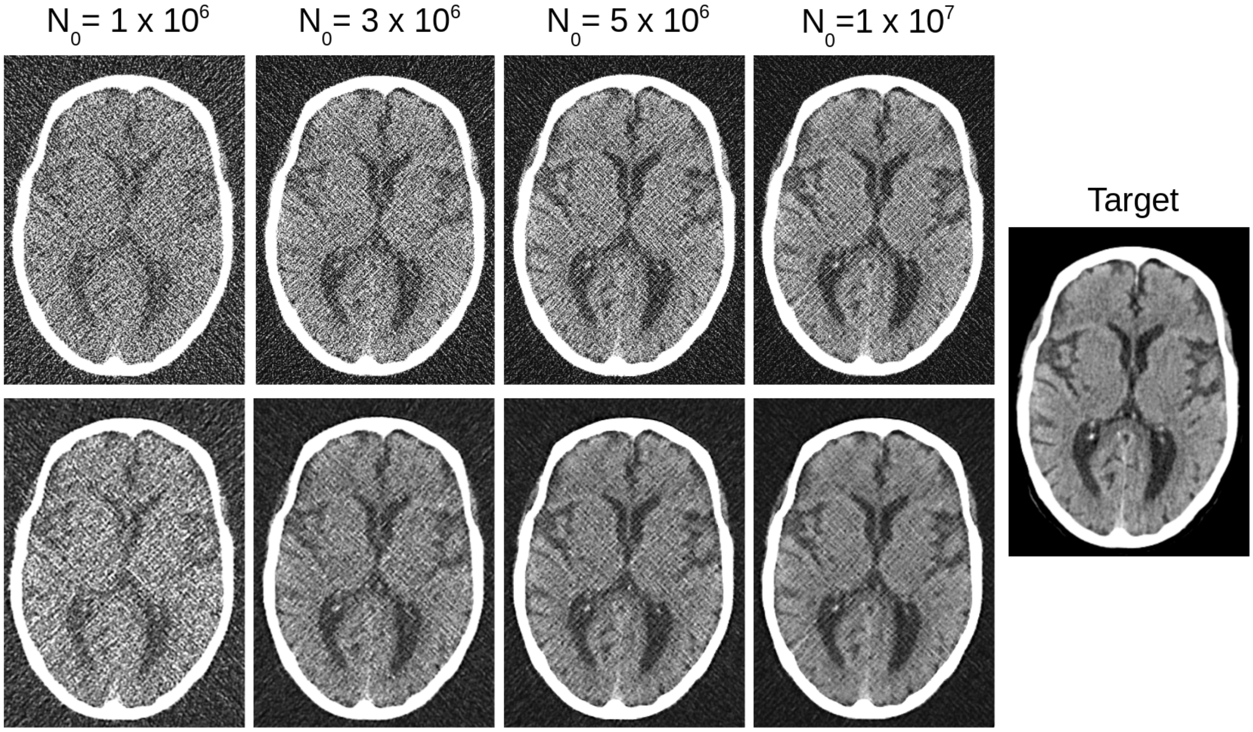


Figure 35: A visual example of the application of simulated noise of varied levels to a target image by changing the amount of influx photons N_0 (top). The corresponding WGAN-VGG (FOLD1 model) denoising output (bottom). The target is a high dose image from the test set of FOLD1.

		$N_0 = 1 \times 10^6$		$N_0 = 3 \times 10^6$		$N_0 = 5 \times 10^6$		$N_0 = 1 \times 10^7$	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
FOLD1	LDCT	62.0591	0.3608	62.1090	0.3654	62.1244	0.3666	62.1390	0.3675
	Denoised	62.0813	0.3667	62.1220	0.3679	62.1347	0.3681	62.1471	0.3684
FOLD2	LDCT	62.0999	0.3662	62.1537	0.3714	62.1699	0.3727	62.1855	0.3738
	Denoised	62.1355	0.3732	62.1681	0.3743	62.1780	0.3745	62.1881	0.3747
FOLD3	LDCT	61.9996	0.3514	62.0527	0.3563	62.0679	0.3575	62.0833	0.3586
	Denoised	62.0458	0.3575	62.0882	0.3588	62.1003	0.3591	62.1124	0.3594
FOLD4	LDCT	62.6055	0.4334	62.6609	0.4393	62.6774	0.4408	62.6930	0.4420
	Denoised	62.5799	0.4410	62.6235	0.4422	62.6366	0.4425	62.6492	0.4427
FOLD5	LDCT	62.0505	0.3594	62.1029	0.3640	62.1183	0.3652	62.1331	0.3662
	Denoised	62.0226	0.3652	62.0644	0.3662	62.0767	0.3664	62.0888	0.3666
Average	LDCT	62.1629	0.3742	62.2158	0.3793	62.2316	0.3806	62.2468	0.3816
		(± 0.0999)	(± 0.0134)	(± 0.1005)	(± 0.0136)	(± 0.1007)	(± 0.0136)	(± 0.1008)	(± 0.0137)
	Denoised	62.1730	0.3807	62.2132	0.3819	62.2253	0.3821	62.2371	0.3824
		(± 0.0926)	(± 0.0137)	(± 0.0931)	(± 0.0137)	(± 0.0932)	(± 0.0137)	(± 0.0934)	(± 0.0137)

Table 7: Quantitative denoising results on various levels of simulated noise from the five models trained on 20 clinical patients. SSIM value of the lowest simulated dose level ($N_0 = 1 \times 10^{-6}$) increased to that of five times higher ($N_0 = 5 \times 10^{-6}$), owing to WGAN-VGG denoising.

4.5 SAGAN Testing and Model Comparison

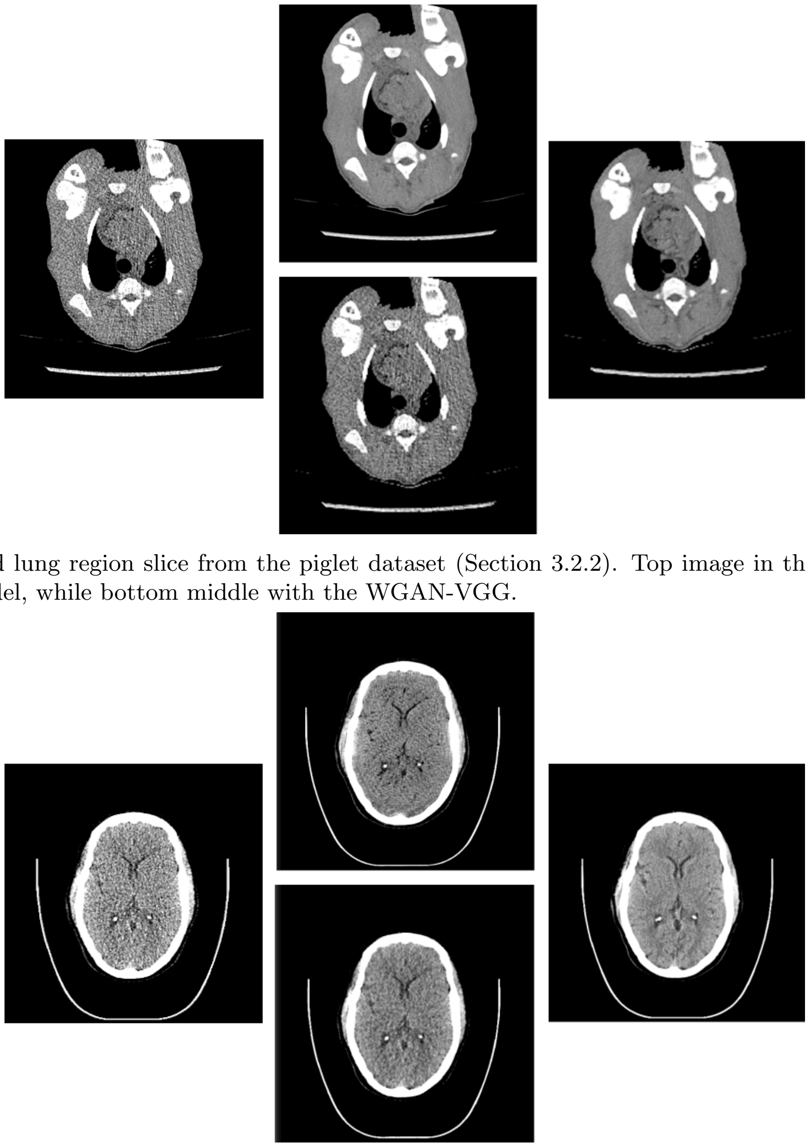
The visual inspection of SAGAN and WGAN-VGG validation on the two different datasets is presented in Figure 36. The associated quantitative results are shown in Table 8. Both models demonstrated some denoising ability, successfully avoiding too much blur in the output, supporting the claims published by *Yi et al.* and *Yang et al.*

Unsurprisingly, SAGAN achieved better numerical improvement and aesthetical appeal of the output on the piglet dataset. However, there exists significant bias in the fact that evaluation is performed on the training data. WGAN-VGG output did not yield significant visual enhancement, but still led to small positive increase in image statistical properties through denoising, irrespective of the fact that model training data consisted of image slices from the human brain region only.

On the other hand, the outcome of the SAGAN network denoising of the previously unseen local clinical patient data did not produce any quantitative improvements. Furthermore, as seen in Figure 36b, the output image not only suffered the change in contrast, but also demonstrated deterioration in the appearance of finer details and overall structure impairment, as compared to both the input and the target images.

	Piglet		Clinical Patient	
	PSNR	SSIM	PSNR	SSIM
LDCT	33.0080	0.7231	34.6686	0.8041
SAGAN	33.8374	0.8168	33.8635	0.7114
WGAN-VGG	33.1396	0.7606	35.6816	0.8083

Table 8: Quantitative output associated with images presented in Figure 36.



(a) Randomly selected lung region slice from the piglet dataset (Section 3.2.2). Top image in the middle was denoised with the SAGAN model, while bottom middle with the WGAN-VGG.

(b) Randomly selected brain slice from the clinical data (Section 3.2.1). Following the example above, top row in the middle demonstrates SAGAN output and bottom WGAN-VGG output. The output from SAGAN appears more distorted than that of WGAN-VGG.

Figure 36: Visual output examples of the denoising result from the two different networks on the two separate datasets, consisting of (left to right): LDCT input images, corresponding model output and target diagnostic dose images. (a) The effective full dose scan comprised 14.14 mSv and LDCT was estimated at 0.71 mSv (5% of the full dose). The SAGAN post-processing images were displayed in the "abdomen" window. (b) The effective high dose images from the local clinical scans amounted to 1.5 mSv and 0.2 mSv for the low dose (13% of the full dose).

5 Discussion

5.1 Outcomes

The presented thesis project resulted in a novel implementation of the WGAN-VGG [6] network on a local clinical dataset of paired low- and high-dose CT images. A learning rate scheduler and early stopping were introduced to the original model by *Yang et al.*. During the fine-tuning of the network, the learning rate and the trade-off between the adversarial and perceptual losses were adjusted for improved performance on the unseen data. The optimal values were found to be α = 1e-6 for the former and $\lambda_1 = 0.15$ for the latter. The final model (precisely - the average across 5 folds) introduced an enhancement in the quality of the denoised images as compared to their low-dose inputs. The numerical improvement was observed in PSNR and SSIM values along with significantly lower noise standard deviations across same uniform regions. The denoised images also scored, on average, higher in visual quality than LDCT in the subjective clinical evaluation. Finally, the implemented model raised the PSNR and SSIM values of the lowest simulated dose images to the dose level five times higher. WGAN-VGG, trained on the clinical data, also generalised better to the unseen pig data, as opposed to SAGAN, trained on the pig and tested on the clinical patients.

5.2 Optimisation Strategies & Performance Evaluation

Network performance optimisation is more intricate than simply finding local minima of some loss function. An appropriate convergence criterion must be chosen to accommodate the task at hand. In the case of a GAN-type network, this implies equilibrium between the generator and the discriminator. Since the output of another network effectively acts as a loss function, searching for model convergence becomes more fleeting and volatile. Hence, implementing a manual stopping criterion becomes important: continuing to train a model that already has a perfectly optimised discriminator causes the generator to learn from a randomised feedback, which results in poorer performance over time [59] (confirmed and demonstrated in Section 4.1.1). Moreover, the appropriate span of training for accurate model fit in the case of WGAN-VGG was additionally largely defined by the size of the training data. It is also important to understand, particularly during the fine-tuning process, that specific model performance is not necessarily a linear function of any single hyperparameter. The model in question, for example, possesses a rather high capacity, allowing for many nonlinear interactions. This might require different optimal training time per given choice of parameters. All of the above issues are easily tackled by the common regularisation strategy of early stopping. In this case, it involved discontinuing training iterations past the point of degradation (or prolonged plateauing) of corresponding perceptual loss (refer to Section 4.1.3). Finally, loss functions for any given model need to be inspected for signs of underfitting (implying poor representation and parameter estimations) and/or overfitting (which is likely to result in weak generalisation) during training. It is important to remember that the best performing models usually have exactly the appropriate amount of bias in a way that guides parameter estimation, without being too dependent on the training data. Whether any given dataset is sufficient for the job can only be determined on an individual basis.

Identifying appropriate model performance measure is of great importance because it is only possible to perform indirect evaluation, be it a loss function or fit accuracy. PSNR is a useful metric in this regard as it is inherently linked to the quality of the generator performance and accompanying noise estimation (and the degradation in the accuracy of PSNR estimation with worsened G performance was indeed observed, for instance, in Figure 31).

The addition of VGG perceptual loss was proven to be beneficial for the visual appearance of the output (Figure 28). Moreover, while GAN-types are overall successful in generating realistic images, the adversarial loss alone does not account for the degree of structural preservation. Therefore, the addition of an auxiliary loss for output-target correspondence is a necessity rather that accessory when diagnostic qualities are concerned.

5.2.1 Hyperparameter Tuning

As had been previously demonstrated in Sections 4.1.2, faster learning rate naturally resulted in accelerated model convergence, while capturing aspects of the noise distribution early on. However, irrespective of the tendency of models with slower learning rate to underfit initially, these yielded the best end results in terms of both the quantitative noise measure and perceived quality of the final image. This result simply justifies the mainly heuristic approaches to finding optimal collections of hyperparameters and regularisation routes for each specific network architectures and data types.

Several models with learning rate schedulers were trained in order to observe if comparable PSNR gain could be achieved more quickly, based on the tendency of a greater α value to achieve a better representation of the data distribution at a faster rate. These models were trained by allowing the network to update faster at initial stages, followed by a steady decrease in the learning rate. This was made in an attempt to rectify model fits in the later training stages by preventing the gradient descent "overshoot" of the true local minima (also a plausible explanation for models favouring slower learning rate) and, hence, the ultimate loss divergence and poor predictions. However, it did not result in any significant development. It might also be rationalised by the choice of the network optimisation algorithm itself, since the utilised Adam optimiser aims to handle appropriate updates of the learning rate through individual tweaks to each parameter. Adam proved to be robust enough with the default rate and simply required appropriate number of iterations for given models to achieve the most successful quantitative and visual enhancement. Lastly, the processing time difference among investigated learning rates proved to be negligible, so there was no massive trade-off in the form of computational cost and time. Any further decrease of the learning rate, however, would be ineffective from a practical point of view. [77] Overall, introducing a learning rate scheduler did not lead to any quantitative improvements, but was still helpful for verifying the optimal initial rate value. On the contrary, an addition of early stopping proved useful, as models trained for an appropriate number of iterations demonstrated better performance.

5.3 WGAN-VGG Denoising Results

The resulting noise distribution plots (Figures 32 and 31) indicated that the network was successful in mimicking the true noise. Additionally, upon visual inspection and according to clinical evaluation, no abnormalities or pathologies were introduced in the generated images.

The generally small variations in the PSNR values across models might be explained by the fact it is encapsulated in the optimisation process directly. Moreover, some deviations in statistical parameters could have been affected by the rounding off in the stages of data pre-/post-processing and writing to different DICOM files. The difference in magnitude of PSNR, as measured by the network directly (for example, Tables 1 and 8) and later on the output DICOM files (for example, Tables 2, 3 and 4), were due to the fact the images were processed in 32-bit, but the output pixels were written to DICOMS as 16-bit integers. Nevertheless, these fluctuations were deemed acceptable for the purpose of the analysis, since it was only concerned with the relative difference, as compared to the target. Therefore, no inconsistencies were observed otherwise.

All statistical folds achieved comparable PSNR and SSIM gains (Appendix A). The resulting mean gain in PSNR and SSIM from a different WGAN-VGG model on the two published images in *Yang et al.* appeared to be 4 times higher that the one achieved in this project for the five image sets in Figure 34 (3.6570 ± 0.0532 and 0.04525 ± 0.0026 for the former against 0.7518 ± 0.2330 and 0.0100 ± 0.0010 from the latter). Alas, the implications of such a comparison are uncertain. Firstly, the former model was trained on 10 patients, as opposed to the proposed trained on 20, potentially making the first more biased. Secondly, the noise distributions learned by respective models are inherently different in nature. Finally, the validity of comparing such metrics across different models and datasets is rather ambiguous (elaborated more in Section 5.6).

When the mean CT numbers of the analysed image regions were concerned (Figure 33), all of the models achieved significant reduction in SD, that tended towards, but did not fall below, the SD values of the respective target images. This is indicative of substantial noise reduction without any over-smoothing in the regions of interest. On the contrary, the resulting mean HU values were either lower or higher than in corresponding low and high dose images, suggesting that some information content was compromised in the process. This might serve as the rationale behind some of the output images appearing slightly different during clinical evaluation.

Overall, the k-fold validation yielded similar results across models, with an exception of PSNR and SSIM reduction in Table 4 and increse in HU values in Table 5 post-denoising in FOLD4 and FOLD5, contrary to other similar outputs. This might imply that these two models generalise slightly worse than the rest, but are still very much similar on average (Appendix A).

5.3.1 Clinical Evaluation

It is important to obtain an opinion from the clinical experts prior to attempting method implementation into clinical practice to ensure the fidelity of the output data. Even though the deep learning denoising method resulted in objective quality improvement, the mean awarded score of the output images was still below 2. Based on the radiologist's assessment, some of the presented model outputs also exhibited a change in perceived structure in comparison to standard LD/HD CT images, evident to an experienced practitioner. This leaves room for improvement on the side of generating perfectly realistic images.

5.4 Simulated Noise

The published results from *Yang et al.* were achieved with a model trained on a real clinical dataset paired with simulated quarter-dose CT images. The x-ray photon measurements, and even electronic noise, can indeed be modelled in accordance with Equation (4). Furthermore, *de Nijs* (2015) [9] demonstrated that for simulating half (or less)-count images Poisson resampling is the method of choice. On the other hand, the study was performed on the SPECT data, which has a different geometry and does not account for the noise distribution in reconstructed CT images, which is not uniform. Thus, while deep learning methods mitigate the uncertainty in noise expression between low and diagnostic dose CT images, the issue still remains in how well a model trained on simulated noise would generalise to real data.

Moreover, attempting to train a WGAN-VGG model with data comprising low dose images from both the simulated and real noise distributions resulted in completely meaningless output (purposefully excluded from the analysis) as, in agreement with *Yang et al.*, the model parameters have to be adjusted or re-trained for different noise properties.

Even though the relation between the true and simulated noise distributions remains an open question, the control of the exact noise level in the simulated images allowed to quantify denoising change with better precision and ensured that the model did not introduce significant changes to images that were already high in PSNR. Furthermore, the denoising output of the lowest dose level yielded SSIM value equivalent to that obtained with five times the number of photons (Table 7). This corresponds to a 20% dose reduction under the assumption the noise is truly representative of the dose level.

5.5 SAGAN Model Comparison

It was hard to assess the general effectiveness of the network performance as the training code had not been made available. A more detailed look at the code could also potentially reveal the reason for the algorithm producing a "target" image identical to that of the corresponding input. Even though the authors claimed that the given model can be applied to a range of anatomies and dose levels, including unseen doses within the training range, there was not enough evidence to support this conclusion.

On the other hand, there is still an obvious upper limit on the allowed dose levels when it comes to real clinical data. Due to the underlying physics, some degree of noise is still present in higher dose images, which could get captured by the generator. While the SAGAN model was not trained on the human patients, the high dose images in the published work enjoyed a much higher dose (14.14 mSv as compared to the clinical analogue of 1.5 mSv). This could also explain more significant visual denoising, as compared to the WGAN-VGG model trained on the clinical data (Figure 36). Finally, it is important to remember that the type of filter employed in reconstruction directly affects noise and blur in resulting images. Thus, consistency across datasets needs to be ensured for a more accurate comparison.

All things considered, the WGAN-VGG model still generalised to the unseen data better that the SAGAN model.

5.6 Limitations

The main challenge posed in the course of the project involved pinpointing the objective measure of noise in images. Still presenting an immense value as a tool for performance evaluation, the overall significance of PSNR as a visual quality metric should be carefully considered. PSNR compares the statistical properties of an image with respect to the ground truth only and becomes more ambiguous when structural preservation and visual appeal are involved, thus telling very little about the overall quality improvements. This is further supported by the small differences in its value across high and low dose images (Tables 1 to 4, Appendix A), while the change as perceived by the human observer is clearly evident (Figure 34, Table 6). Furthermore, since PSNR is measured in terms of relative pixel intensities, insignificant aspects, like a few bright pixels, can jeopardise the validity of the metric altogether. There appears to be a consensus on the fact that SSIM is a better candidate for objective image structural quality estimations, since it is a more intricate measure in comparison to the absolute error of PSNR and MSE, by definition. [78] Nevertheless, the quantitative results from this analysis indicated than an increase in PSNR is usually accompanied by an increase in SSIM. Furthermore, a supplement of a third noise-quantifying metric in the form of mean CT numbers and their standard deviations in similar regions of the output images should serve as a conclusive argument in favour for observed statistical improvements introduced by the denoising model, as compared to their original low-dose counterpart. Moreover, because the PSNR is analogous to MSE error (of pixel intensities), network architectures that aim to minimise MSE loss were proven to land more substantial quantitative gains [6], while the proposed model focuses instead on the clinically-relevant objective of preserving structure and limiting blurring.

Irrespective of being sufficient for the numerical analysis within the scope of the project, the debatable validity of the quantitative metrics involved from a global perspective also raises concerns about the fidelity of comparing achieved quantitative results with similar publications. Furthermore, the concerns for using PSNR and SSIM alone for medical image quality evaluation has been similarly expressed by the authors of presently utilised network architecture who noted that direct iterative reconstruction methods yielded the highest PSNR and SSIM values, while also introducing artefacts. [6] Perhaps, the most important aspect to consider in this matter is the fact models from *Yang et al.* were trained on simulated low-dose noise.

Another inherent limitation aspect comprised the availability of the training data. Firstly, the mandatory requirement of the presented network for spatially-aligned high and low dose images greatly constrained the choice of suitable data, while the improvements that a larger patient cohort can introduce are evident. Secondly, the architecture tends to overfit the data when the training data set is small (as seen, for example, by comparing the generator loss of the network trained on 10 patients in Figure 25 with that trained on 22 in Figure 32). The fidelity of the generated images also appeared to greatly improve for the latter (for instance, through inspecting the respective outputs in Figures 26a with Figures 30 and 28). Finally, the abundance of training data is generally associated with more successful network training, irrespective of the network architecture. [52]

Some intrinsic properties of the data itself could also serve as potential source of confusion for the generator network or or add to the ambiguity of the quantitative metric. For instance, the tiny spatial misalignments and inconsistencies across paired image series that cannot be helped or the ratio of grey to white matter, which could have had an effect on learning or estimations of the noise standard deviation.

Lastly, the fact that the hyperparameter tuning was done with only 2 validation patients could lead to poorer generalisation. Nevertheless, the decision was made in favour of maximising the training cohort instead, as the total amount was limited due to ceased patient scanning during national lockdown. Training models with a consistent data size over the course of the project was also complicated, because the high dose scans and appropriate reconstruction were not always made available. Nonetheless, the size of the training cohort utilised in this project did not vary much from comparable publications (for example, *Yi et al.* [6] trained the original WGAN-VGG model on 10 patients).

5.7 Perspectives

Regularising and adjusting the model to the utilised dataset involved a long process of various model training, which was otherwise eased by the WGAN-VGG architecture, making performance monitoring a straight-forward task. The model itself demonstrated good denoising ability and quality improvement, but the generated image structure still remained suboptimal for clinical purposes. Apart from accessing more training data, it could prove useful to investigate other GAN architectures in the future, for example more complex generators, like that in [21]. Furthermore, there exist alternative GAN architectures that do no require paired data for training (for instance, *Wolterink et al.* (2017) [20] used the adversarial loss only and *Kang et al.* (2018) [23] employed a cyclic GAN for for LDCT denoising).

Another way to ease the necessity for paired clinical data might involve investigating more elaborate approaches to simulating noise in CT. It would be intriguing to quantify the extent of the influence of the electronic noise on the overall noise distribution along with relative fidelity of simulated datasets. Furthermore, as variations in the noise levels are associated with, among other things covered above, reconstruction methods themselves, it might prove beneficial to explore the denoising methods directly in the sinogram domain. This approach also addresses the irreversible information loss problem associated with post-reconstruction methods. There also exist potential strategies to avoid using paired data entirely as in *Zhu et al.* (2017) [45].

Moreover, a possible collaboration with the local forensic department became a topic for discussion in the course of the project. The great advantage of the post-mortem CT scans is the potential for an abundance of paired images with a less strict upper dose threshold.

There is a plethora of currently available and actively utilised deep learning methods in the medical imaging field. For instance, GE proposed the *TrueFidelity*, a neural-network based CT image reconstruction technology that could challenge both the FBP and IR methods. [13] Presently, implementing a post-processing method for improving constitution of a gold standard reconstruction output is likely to assist the diagnostic process for radiologists in the future.

Finally, there is nothing that suggests the application of the presented denoising method is limited to a single imaging modality.

6 Conclusion

Ultimately, the proposed model has the capacity for visually significant LDCT image denoising without introducing blur or oversmoothing in the generated image. A variety of approaches to evaluating the denoising results should serve as sufficient evidence in favour of hypothesised improvement. Presently, the goal consisted mainly of upscaling the quality of existing low dose images with the aim of assisting the diagnostic process in the future. Given the relatively small training patient cohort, a statistically significant improvement was still observed in both the numerical output and the radiologist's assessment. Accessing more training data will lead to further advancements in the model performance and continued research will reveal more effective approaches to the gentle balancing act of eradicating noise and conserving finer structure for diagnostic purposes.

It is important to recognise that currently, the generated images could not replace the diagnostic dose. But improving the quality of the low dose is both feasible and promising. Furthermore, training a robust model validated on a variety of unseen data would allow for scanning patients at lower dose levels, while achieving image quality comparable to that obtained with conventional CT.

The main advantage of the presented work is in the unique real clinical data employed for training as well as a larger number of patients than in *Yang et al.*. Thus, the resulting trained model parameters should be better adapted to realistic noise distribution in CT. This is especially relevant for the proposed WGAN-type, as the adversarial loss in this case aims to minimise the difference between the respective probability distributions.

In conclusion, two compelling questions still remain open for debate: whether the need for paired training data could be suppressed in the future and whether the gap between the network output and the golden standard could be breached.

7 Acknowledgements

I am incredibly grateful for the commitment of all my supervisors to this thesis project and me in my capacity as an MSc student. I would like to thank (in no particular order) Flemming Littrup Andersen, Claes Nøhr Ladefoged and Sofie Lindskov Hansen for always finding time to facilitate my queries, as well as all the advice and the invaluable insight into the deep learning world, programming skills and CT technology that I had acquired in the course of this work. I greatly appreciate Anne Kiil Berthelsen, MD and Chief Physician, for taking time out of her busy schedule to assist with the clinical evaluation of the output images. I am also indebted to Adam Espe Hansen, who had introduced me to the group and, by all means, my primary supervisor Lise Arleth and the censor Søren Baarsgaard Hansen, without whom this project would not be possible. Finally, I would like to acknowledge everyone at the Clincally Applied Artificial Intelligence group at Rigshopitalet for the most excellent and stimulating conversations and being a pleasure to share the office with.

I would like to dedicate this work to my beloved grandfather, who passed away on the day of the submission. I would not have made it this far without him, and I know I made him proud.

References

[1] Flower, M. A. (2012). Webb's Physics of Medical Imaging, 2^{nd} Edn, *CRC Press, Taylor & Francis Group*; pp. 409-504, 509, 512-513. ISBN: 9781466568952.

[2] Brenner, D. J. & Hall, E. J. (2007). Computed tomography - an Increasing Source of Radiation Exposure, *New England J. Med.*, 357(22); pp. 2277–2284. doi: 10.1056/NEJMra072149.

[3] Shrimpton P. C. & Edyvean, S. (1998). CT Scanner Dosimetry, *Br J Radiol.*; 71(841); pp. 1-3. doi: 10.1259/bjr.71.841.9534691.

[4] Geyer, L. L. et al. (2015). State of the Art: Iterative CT Reconstruction Techniques, *Radiology*, 276(2); pp. 339–357. doi: 10.1148/radiol.2015132766.

[5] Dai, H. et al. (2018). Limited-View Cone-Beam CT Reconstruction Based on an Adversarial Autoencoder Network With Joint Loss, *Conf Proc IEEE Eng Med Biol Soc*; pp. 5570-5574. doi: 10.1109/EMBC.2018.8513659.

[6] Yang, Q. et al. (2018). Low Dose CT Image Denoising Using a Generative Adversarial Network with Wasserstein Distance and Perceptual Loss, *IEEE Trans. Med. Imaging.*, 37(6); pp. 1348-1357. doi: 10.1109/TMI.2018.2827462.

[7] Chen, H. et al. (2017). Low-Dose CT via Deep Neural Network, *Biomed Opt Express*, 8(2); pp. 679–694. doi: 10.1364/BOE.8.000679.

[8] Wang, J. et al. (2005). An Alternative Solution to the Non-Uniform Noise Propagation Problem in Fan-Beam FBP Image Reconstruction, *Med Phys*, 32(11); pp. 3389-3394. doi: 10.1118/1.2064807.

[9] Robin de Nijs (2015). Comment on: 'A Poisson resampling method for simulating reduced counts in nuclear medicine images', *Phys. Med. Biol. 60 5711*, 60(14). doi: 10.1088/0031-9155/60/14/5711.

[10] Li, G. et al. (2016). A Noise Power Spectrum Study of a New Model-Based Iterative Reconstruction System: Veo 3.0, *J Appl Clin Med Phys*, 17(5); pp. 428-439. doi: 10.1120/jacmp.v17i5.6225.

[11] Qi, J. (2003). Noise Propagation in Iterative Reconstruction Algorithms with Line Searches. *IEEE Cat. No.03CH37515*, 4. doi: 10.1109/NSSMIC.2003.1352406

[12] Von Falck, C. et al. (2013). Influence of Sinogram Affirmed Iterative Reconstruction of CT Data on Image Noise Characteristics and Low-Contrast Detectability: An Objective Approach, *PLoS One*, 8(2): e56875. doi: 10.1371/journal.pone.0056875.

[13] Hsieh, J. et al (2019). A new era of image reconstruction: TrueFidelity, *General Electric Company, GE Healthcare* (JB68676XX).

[14] Krizhevsky, A. et al. (2012). Imagenet classification with deep convolutional neural networks, *Advances in neural information pro- cessing systems*; pp. 1097–1105. doi: 10.5555/2999134.2999257.

[15] Hof, R. D. (2013). 10 Breakthrough Technologies - Deep Learning. *MIT Technology Review*, Cambridge, MA, USA.

[16] Greenspan, H. et al. (2016). Guest Editorial. Deep Learning in Medical Imaging: Overview and Future Promise of an Exciting New Technique, *IEEE Trans Med Imaging*, (35); pp. 1153-1159. doi: 10.1109/TMI.2016.2553401.

[17] Wang Ge (2016). A Perspective on Deep Imaging. `arXiv:1609.04375`.

[18] Litjens, G. et al. (2017). A survey on deep learning in medical image analysis. `arXiv:1702.05747`.

[19] Yi, X. & Babyn P. (2018). Sharpness-Aware Low-Dose CT Denoising Using Conditional Generative Adversarial Network, *J Digit Imaging*, 31(5); pp. 655-669. doi: 10.1007/s10278-018-0056-0.

[20] Wolterink, J. M. et al. (2017). Generative Adversarial Networks for Noise Reduction in Low-Dose CT, *IEEE Trans. Med. Imaging.*, 36(12); pp. 2536-2545. doi: 10.1109/TMI.2017.2708987.

[21] Kang, E. et al. (2018). Deep Convolutional Framelet Denosing for Low-Dose CT via Wavelet Residual Network, *IEEE Trans. Med. Imaging.*, 37(6); pp. 1358-1369. doi: 10.1109/TMI.2018.2823756.

[22] Chen, H. et al. (2017). Low-Dose CT with a Residual Encoder-Decoder Convolutional Neural Network (RED-CNN), *IEEE Trans. Med. Imaging.*, 36(12); pp. 2524 - 2535. doi: 10.1109/TMI.2017.2715284.

[23] Kang, E. et al. (2018). Cycle-Consistent Adversarial Denoising Network for Multiphase Coronary CT Aangiography, *Med. Phys.*, 46(2); pp. 550-562. doi: 10.1002/mp.13284.

[24] Podgorsak, E. B. (2005). Radiation Oncology Physics: A Handbook for Teachers and Students, *IAEA, STI/PUB/1196*; Sec. 1.4. ISBN: 9201073046. Available via: `https://www-pub.iaea.org/mtcd/publications/pdf/pub1196_web.pdf`.

[25] Dance, D.R. et al. (2014). Diagnostic Radiology Physics – A Handbook for Teachers and Students, *IAEA*. ISBN: 978921310101.

[26] Podgorsak, E. B. (2006). Radiation Physics for Medical Physicists, *Springer*. ISBN: 9783540250418.

[27] Bushberg, J. et al. (2011). The Essential Physics of Medical imaging, 3$^{\text{nd}}$ edn. (1994), *Lippincott Williams & Wilkins*, Philadelphia; Chapter 10. ISBN: 9780781780575.

[28] Danad, I. et al. (2015). New Applications of Cardiac Computed Tomography: Dual-Energy, Spectral, and Molecular CT Imaging, *JACC: Cardiovascular Imaging*, 8(6). doi: 10.1016/j.jcmg.2015.03.005.

[29] Knoll, G. F. (2000). Radiation Detection and Measurement, 3$^{\text{rd}}$ edn., University of Michigan, *John Wiley and Sons, Inc.*, New York; Chapters 1, 2, 3. ISBN: 9780470131480.

[30] Krane, K. S. (1998). Introductory Nuclear Physics, Revised edition of: Introductory nuclear physics/David Halliday, 2$^{\text{nd}}$ edn. (1955), *John Wiley and Sons, Inc.*, New York; Chapter 7. ISBN: 9780471805533.

[31] Zhang, H. et al. (2014) Statistical image reconstruction for low-dose ct using nonlocal means-based regularization. *Computerized Medical Imaging and Graphics*, 38(6); pp. 423–435. doi: 10.1016/j.compmedimag.2014.05.002.

[32] Von Schulthess, G. K. (2015). Molecular Anatomic Imaging: PET/CT, PET/MR and SPECT CT, 3$^{\text{rd}}$ edn., *Wolters Kluwer Health*. ISBN: 9781451192667.

[33] Cember, H. and Johnson, T.E. (2008). Introduction to Health Physics, 4th Edn, *McGraw-Hill Education*. ISBN: 9780071423083.

[34] ICRP (2007). The 2007 Recommendations of the International Commission on Radiological Protection, *ICRP Publication 103*, Ann. ICRP 37 (2-4). ISBN: 9780702030482. Text available via: `https://journals.sagepub.com/doi/pdf/10.1177/ANIB_37_2-4`.

[35] Åkerblom, G. et al. (2000). Naturally Occurring Radioactivity in the Nordic Countries - Recommendations, *The Radiation Protection Authoritiesin Denmark, Finland, Iceland, Norway and Sweden*. ISBN: 9189230000. Text available via: `https://www.gr.is/wp-content/media/2013/07/NaturallyOccurringRadioactivity.pdf`.

[36] Mazrani, W. et al. (2007). The radiation burden of radiological investigations, *Archives of disease in childhood*, 92(12); pp. 1127–1131. doi: 10.1136/adc.2006.101782.

[37] The Nobel Prize in Physiology or Medicine 1979 . `https://www.nobelprize.org/prizes/medicine/1979/summary/`.

[38] Prince, J. L. and Links, J. M. (2015). Medical Imagining Signals and Systems, 2$^{\text{nd}}$ edn. (2006), *Pearson Education, Inc.*; Chapter 6. ISBN: 9780132145183.

[39] Hobbie, R. K. and Bradley, J. R. (2015). Intermediate Physics for Medicine and Biology, *Springer International Publishing*, Switzerland; Chapter 12. doi: 10.1007/978-3-319-12682-1.

[40] Ioffe, S. & Szegedy C. (2015). Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. `arXiv:1502.03167`.

[41] Delić, V. et al. (2019). Speech Technology Progress Based on New Machine Learning Paradigm, *Comput Intell Neurosci.*; 2019: 4368036. doi: 10.1155/2019/4368036.

[42] Chalapathy, R. & Chawla, S. (2019). Deep Learning for Anomaly Detection: A Survey. `arXiv:1901.03407`.

[43] Rao, Q. & Frtunikj, J. (2018). Deep Learning for Self-Driving Cars: Chances and Challenges, *2018 IEEE/ACM 1st International Workshop on Software Engineering for AI in Autonomous Systems (SEFAIAS)*, Gothenburg; pp. 35-38. `http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8452728&isnumber=8452713`.

[44] Fukushima, K. & Miyake, S. (1982). Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition, *Competition and cooperation in neural nets, Springer, vol. 45*; pp. 267–285. ISBN: 9783642464669.

[45] Zhu, J.Y. et al. (2017). Unpaired image-to-image translation using cycle-consistent adversarial networks. `arXiv:1703.10593`.

[46] Xue, Y. et al. (2018). SegAN: Adversarial Network with Multi-Scale L_1 Loss for Medical Image Segmentation, *Neuroinformatics*, 16(3-4); pp. 383–392. doi: 10.1007/s12021-018-9377-x.

[47] Shin, H. et al. (2018). Medical Image Synthesis for Data Augmentation and Anonymization using Generative Adversarial Networks. `arXiv:1807.10225`.

[48] Frid-Adar, M. et al. (2018). GAN-based Synthetic Medical Image Augmentation for increased CNN Performance in Liver Lesion Classification. `arXiv:1803.01229`.

[49] Ledig, C. et al. (2017). Photo-realistic single image super-resolution using a generative adversarial network. `arXiv:1609.04802`.

[50] Armanious, K. et al. (2018). Retrospective correction of Rigid and Non-Rigid MR motion artefacts using GANs. `arXiv:1809.06276`.

[51] Mitchell. T. M. (1997). Deep Learning, *McGraw-Hill, Inc.*, New York, NY, USA. ISBN: 9780070428072.

[52] Goodfellow, I. J. et al. (2016). Deep Learning, *MIT Press*. `http://www.deeplearningbook.org`.

[53] Guarnieri, R. & P. et al. (2006). Solar radiation forecast using artificial neural networks in south Brazil. *8 ICSHMO*, Foz do Iguaçu, Brazil, INP; pp. 1777-1785. Available in full via: `https://www.researchgate.net/publication/229036664_Solar_radiation_forecast_using_artificial_neural_networks_in_south_Brazil`.

[54] Jain, P. (2019). Complete Guide of Activation Functions: A practical guide comparing advantages, problems, and solution of activation function. Online article accessed 06/04/2020 via: `https://towardsdatascience.com/complete-guide-of-activation-functions-34076e95d044`.

[55] Torrey, L. & Shavlik J. (2009). Transfer Learning, *Handbook of Research on Machine Learning Applications*, IGI Global. doi: 10.4018/978-1-60566-766-9.ch011.

[56] Kingma, D. P. & Ba, J. L. (2017). Adam: A Method for Stochastic Optimization. `arXiv:1412.6980v9`.

[57] LeCun, Y. & Bengio, Y. (1998). Convolutional Networks for Images, Speech, and Time Series, *The Handbook of Brain Theory and Neural Networks*, MIT Press, Cambridge, MA, USA; pp. 255-258. ISBN:0262511029.

[58] Ronneberger, O. et al. (2014). U-Net: Convolutional Networks for Biomedical Image Segmentation. `arXiv:1505.04597`.

[59] Goodfellow, I. (2016). NIPS 2016 Tutorial:Generative Adversarial Networks. `arXiv:1701.00160`.

[60] Goodfellow, I. J. et al. (2014). Generative Adversarial Nets; Ch. 1, 5, 9. `arXiv:1406.2661`.

[61] Yi, X. et al. (2019). Generative Adversarial Network in Medical Imaging: A Review, *Med Image Anal*, 58. doi: 10.1016/j.media.2019.101552.

[62] Isola, P. et al. (2016). Image-to-Image Translation with Conditional Adversarial Networks. `arXiv:1611.07004`.

[63] Theis, L. et al. (2016). A Note on the Evaluation of Generative Models. `arXiv:1511.01844`.

[64] Shi, J. et al. (2014). Blur detection dataset. `http://www.cse.cuhk.edu.hk/~leojia/projects/dblurdetect/dataset.html`

[65] Yi, X. & Eramian, M. (2016). LBP-Based Segmentation of Defocus Blur. *IEEE Trans Image Process*, 25(4); pp. 1626-1638. doi: 10.1109/TIP.2016.2528042.

[66] Gulrajani, I. et al. (2017). Improved Training of Wasserstein GANs. `arXiv:1704.00028`.

[67] Arjovsky, M. et al. (2017). Wasserstein GAN. `arXiv:1701.07875`

[68] Simonyan, K. & Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. `arXiv:1409.1556`.

[69] Emond, P. et al. (2008). PE2I: A Radiopharmaceutical for *In vivo* Exploration of the Dopamine Transporter, *CNS Neuroscience & Therapeutics*, 14; pp. 47-64. doi: 10.1111/j.1527-3458.2007.00033.x.

[70] Jakobson Mo et al. (2018). Dopamine transporter imaging with[18F]FE-PE2I PET and [123I]FP-CIT SPECT — a clinical comparison, *EJNMMI Research*, 8(100). doi: 10.1186/s13550-018-0450-0.

[71] Xia, T. et al. (2012). Ultra-low dose CT attenuation correction for PET/CT. *Phys Med Biol.*, 57(2); pp. 309-28. doi: 10.1088/0031-9155/57/2/309.

[72] Yang, Q. (2018). Low Dose CT Image Denoising Using a Generative Adversarial Network with Wasserstein Distance and Perceptual Loss. Accessed Oct 2019 (last commit Dec 2018) via: `https://github.com/hyeongyuy/CT-WGAN_VGG_tensorflow`.

[73] Rodríguez, J.D. & Pérez, I. (2010). Sensitivity Analysis of k-Fold Cross Validation in Prediction Error Estimation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(3; pp. 569-557. doi: 10.1109/TPAMI.2009.187.

[74] Yi, X. (2017). Sharpness-Aware Low-Dose CT Denoising Using Conditional Generative Adversarial Network. Accessed Sept 2019 (last commit Jan 2019) via: `https://github.com/xinario/SAGAN`.

[75] Wang, Z. et al. (2004). Image Quality Assessment: From Error Visibility to Structural Similarity. *IEEE Trans Image Process.*, 13(4); pp. 600-612. doi: 10.1109/tip.2003.819861.

[76] Originally by Peter Neelin (1992). Software repository: `http://bic-mni.github.io/` (last accesssed: 29/05/2020).

[77] Bengio, Y. (2012). Practical recommendations for gradient-based training of deep architectures. `arXiv:1206.5533`.

[78] Zhang, L. et al. (2012). A comprehensive evaluation of full reference image quality assessment algorithms. *19th IEEE International Conference on Image Processing*; pp. 1477-1480. doi: 10.1109/ICIP.2012.6467150.

A Quantitative Results Associated with the Test Images from the 5 WGAN-VGG Cross-Validation Models

	PSNR		SSIM	
	LDCT	Output	LDCT	Output
FOLD1	81.8250 ± 0.8096	82.3864 ± 0.8710	0.9643 ± 0.0017	0.9698 ± 0.0017
	66.9681 ± 0.2109	66.9885 ± 0.2117	0.7623 ± 0.0074	0.7661 ± 0.0077
	80.2639 ± 1.1345	80.6742 ± 1.1892	0.9432 ± 0.0034	0.9483 ± 0.0032
	67.1800 ± 0.1809	67.1839 ± 0.1803	0.7522 ± 0.0066	0.7538 ± 0.0065
	68.0484 ± 0.2865	69.5917 ± 0.3572	0.8014 ± 0.0082	0.8231 ± 0.0097
FOLD2	77.4451 ± 0.8253	77.6138 ± 0.8915	0.9202 ± 0.0056	0.9241 ± 0.0057
	84.1317 ± 0.8558	85.7560 ± 1.0772	0.9707 ± 0.0016	0.9796 ± 0.0013
	85.3825 ± 0.7487	86.5430 ± 0.9349	0.9771 ± 0.0014	0.9828 ± 0.0014
	66.9131 ± 0.1806	66.9538 ± 0.1826	0.7747 ± 0.0068	0.7783 ± 0.0066
	80.1323 ± 1.1757	80.9534 ± 1.3388	0.9381 ± 0.0039	0.9442 ± 0.0036
FOLD3	73.6664 ± 1.1523	74.2485 ± 1.2999	0.8579 ± 0.0075	0.8609 ± 0.0076
	67.2408 ± 0.1872	67.2712 ± 0.1853	0.8152 ± 0.0056	0.8184 ± 0.0057
	81.8121 ± 0.8040	82.7601 ± 0.9540	0.9618 ± 0.0017	0.9689 ± 0.0015
	66.4992 ± 0.1890	66.5311 ± 0.1883	0.7544 ± 0.0067	0.7571 ± 0.0066
	84.8116 ± 0.9788	86.7181 ± 1.2372	0.9667 ± 0.0025	0.9748 ± 0.0022
FOLD4	78.7799 ± 0.8722	78.8583 ± 0.8759	0.9328 ± 0.0040	0.9390 ± 0.0040
	75.1007 ± 1.0758	75.6016 ± 1.2111	0.8967 ± 0.0060	0.8985 ± 0.0058
	87.5983 ± 0.7987	88.5547 ± 0.9812	0.9822 ± 0.0011	0.9867 ± 0.0010
	89.9497 ± 0.8303	91.0401 ± 1.0075	0.9839 ± 0.0011	0.9886 ± 0.0010
	66.9581 ± 0.2107	66.9370 ± 0.2118	0.7526 ± 0.0074	0.7524 ± 0.0073
FOLD5	72.8889 ± 1.2640	73.5789 ± 1.4353	0.8399 ± 0.0078	0.8423 ± 0.0078
	82.5099 ± 1.1286	83.2858 ± 1.3658	0.9522 ± 0.0031	0.9536 ± 0.0031
	89.6671 ± 0.7975	90.8500 ± 0.9647	0.9847 ± 0.0011	0.9900 ± 0.0008
	82.1655 ± 1.0126	83.0797 ± 1.1793	0.9532 ± 0.0028	0.9597 ± 0.0026
	76.4294 ± 1.3545	76.8996 ± 1.4597	0.8624 ± 0.0091	0.8651 ± 0.0092