POLITECNICO
DI TORINO

Master degree course in Biomedical Engineering

Master Degree Thesis

Anatomically constrained Cross-domain CT image translation using CycleGAN

Supervisor
Prof. Filippo Molinari

Candidate
Francesco Maso

Co-Supervisors
Prof. Pietro Gori
Prof. Isabelle Bloch
Dott. Giammarco La Barbera

March 2021

Acknowledgements

I would like to acknowledge my family who supported me during my studies in Turin and in France.
I would also like to thank the Polytechnic of Torino for giving me the opportunity to live this incredible experience, despite this particular period.
I would also like to express my appreciation to Telecom Paris for welcoming me and providing all the help and support I needed. I thank all the PhD students and professors I met, who immediately created a very friendly and stimulating working environment.
A special thanks goes to Professor Gori, who from the beginning was extremely kind and helpful in explaining and solving any doubts or technical problems.
I would also like to thank Giammarco in particular for the enormous technical and moral support he has shown in these months, during the period in France and during the smart working from home.
Last but not least, I would like to thank Chiara for pushing me to take this experience, and without whom I would not be here today.

Contents

Introduction

This study describes a method to perform an image to image translation by means of a deep generative model. Its purpose is to obtain, from abdomen CT images with no contrast, contrast enhanced CTs and vice versa.
The entire study was carried out through a six-month internship in Paris at the Ecole Nationale Supérieure des Télécommunications (TELECOM Paris), in the department of Image, Data, Signal (IDS). The medical images are obtained from different databases that are available online, and they specifically analyse only the abdominal component of patients, from the lungs to the femoral head.
In particular, the generative model is composed of Cycle Consistent Adversarial Networks (called CycleGAN), whose main properties are based on the recent model described by Zhu et al. [1]. Where they describe two generators, based on deep neural networks, able to generate new images starting from two unpaired sets of images with different characteristics.
One of the main problems related to medical imaging is the small number of images in the databases, provided by hospitals or available online in open-access, which is usually not enough to accurately train deep learning methods. The lack of large data-sets in medical imaging, with respect to other branches of Computer Vision, is due to the fact that acquiring medical images is usually expensive and difficult and that in many cases images can not be distributed in open-access for privacy reasons. To help identify pathologies, doctors often inject a contrast agent, e.g. iodine or barium-based, which is absorbed and then evacuated by the body. Injected images better highlight the interface between different tissues and structures (e.g., kidney/tumor), facilitating the identification of possible pathologies in the patient and are therefore usually preferred to non-injected images for segmenting anatomical organs for surgery planning or research purposes. Unfortunately, however, for many patients the use of contrast agents is not always viable due to the possiblility of complications: such as allergic reactions or chronic pathologies [2]. This obviously makes it more difficult to diagnose the disease and limits the production of certain types of images. For this reason, this study describes a method where we try to artificially generate medical images with the contrast, even though it had never been injected into the patient. In particular, CT images can be improved through

the insertion of contrast agent in different phases: in this study we focused on images acquired in the early-arterial phase. At the same time we apply the inverse transformation, obtaining contrast-free CT images, which can be used to increase the number of available data sets with images that have an excellent ground truth. The translation is done by using only healthy patients, to we prevent the generation of artefacts of any pathologies, such as thrombosis or stenosis. These are only visible in the images with contrast inserted, because in the image without contrast there is greater homogeneity between different anatomical structures. Hence the generation of pathological elements may be difficult for a doctor to interpret. The use of databases with patients with pathologies may be the baseline for a future study.
The first chapter briefly describes the type of medical image used in this study; the technology with which it is obtained and its main characteristics. The importance of the use of contrast agents for medical treatment is also explained in more detail, as well as the main problems that may prevent their use.
The second chapter describes the main instruments used for the study: starting from the theory behind machine learning, up to the characteristics and advantages of Deep learning, where some of the main neural network models and generative models currently used, such as the GAN and CycleGAN, are described.
In the third chapter, the most recent studies, where a Generative Adversarial Network (and its variants) has been used to obtain new medical images, are detailed. Particular attention is given to the CycleGAN, which seems to be well suited for the goal of this project.
In the fourth chapter, we describe the employed data-sets and all the pre-processing steps we proposed.
The fifth chapter describes in details the proposed method which is based on the well-known CycleGAN architecture. We highlight the main features and different steps to improve the quality of the data generated.
The sixth chapter describes the methodology with which the results obtained were evaluated, from a qualitative and quantitative point of view, and the parameters with which they were evaluated. In the final part, comments on the implemented generative model are elaborated from the results acquired, describing possible future implementations with this technology and also analysing new possible improvements that can be applied to the study.

Chapter 1

Biomedical Aspects

First of all, this chapter describes the *Computed Tomography* and how it is able to produce medical images for diagnostic applications, which were the fundamental data on which the entire study was built. It is then described how the contrast agent act and how it promote the visualization of biological tissues, providing biochemical information on different organs and their performance. In the third and final section, the use of CT images to diagnose abdominal organ pathologies is described.

1.1 Computed Tomography

Computed Tomography (also known simply as CT) is a diagnostic imaging technique. It was born and diffuses thanks to its capacity to overcome the limitations of traditional radiography. CT in particular, allows to discriminate tissues, with very close attenuation coefficients and to display 3D volumes with an high resolution.
The principle for the calculation of the attenuation coefficients is based on Lambert-Beer's law, which relates the variation in the number of photons belonging to radius X, after hitting a material, with a specific linear attenuation coefficient:

$$N = N_0 e^{-\mu x} \tag{1.1}$$

Where N_0 represents the number of photons striking the material, while N represents the attenuation after they have passed through it. The attenuation coefficient is indicated by μ, and x represents the space covered in the material.
The image is obtained with an X-ray tube, that can produce a beam of X-rays and a group of detectors, whose disposition and number can vary depending on the generation of the machine, which are used to deduce the density of the tissues affected by the photons of the X-rays. In fact, at the beginning the machines were composed of a series of collimators aligned with each other, always positioned in opposition to the photon beam emitters. Subsequently it was decided to keep only

the X-ray tube moving and to create a ring with detectors [3].
Thanks to a technology called slip ring, the X-ray tube takes power directly from the circular track on which it rotates, minimising resistance and the friction to reduce electromagnetic disturbances as much as possible. Moreover, thanks to the realization of multisclices scanners, it was possible to obtain more images of the patient at the same time, in this way the exposure time is reduced.
Using an angular sampling, each volumetric element of the body section analysed is irradiated from different angles, successively its radio density is calculated through a Filtered back projection algorithm. In this way it will be possible to obtain the desired 3D volume. This strategy is necessary to overcome the problem of identifying tissue layers covered by others with a higher absorption coefficient.
The average thickness of each slice of the patient is approximately 0.5 mm, while the typical resolution is 512 x 512 pixels (or 1024 x 1024).
The density of different biological tissues is calculated using a linear attenuation coefficient expressed in Hounsfield units (HU).

$$\mu(HU) = 1000\frac{\mu - \mu_{H_2O}}{\mu_{H_2O}} \tag{1.2}$$

Where μ represents the linear attenuation coefficient of the hit tissue, while μ_{H_2O} represents the linear attenuation coefficient of the water. This densitometric scale is adimensional and traditionally includes 2000 values, starting from a minimum of -1000, the region of ray transparent. While in the higher extreme we find highly radiopaque elements such as bone (0 HU corresponds to water density) [4].
In order to display the image in the best possible way there is the possibility to select the range, which takes the name of window (Fig. 1.1), of grey values that can be displayed. So it is possible to observe only some tissues and discriminate biological elements that have very similar attenuation values. Anatomical districts such as the chest, for example, need different windows in order to correctly recreate the entire volume; since there are lungs and bone components that exhibit different levels of radiopacity.

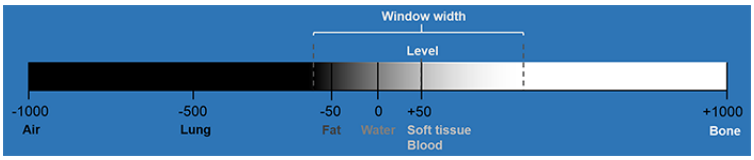


Figure 1.1: A Window Width for soft tissue. Adapted from [5].

In conclusion, the quality of the volume acquired by CT is also influenced by the scanning time, which depends on the rotation speed of the X-ray tube and the number of slices acquired. In particular,the *pitch factor* measures the ratio between the number of rotations acquired and the gap between them.

Obviously a slower acquisition leads to a higher quality of images, but the collateral effect is to expose the patient to a higher dose of radiation.

1.2 Computed Tomography contrast agents

The introduction of contrast agents for Computed Tomography has allowed a considerable increase in this diagnostic technique, thanks to an improved visualization of the anatomical components of interest, like the interface between two different tissues (e.g., liver/tumor) and making easier the identification of eventual pathologies. However, other imaging techniques, like MRI, are considered safer because they do not use ionizing radiations. Contrast materials to be used in clinical trials must satisfy several requirements before [6]:

- The contrast and its metabolites have to not be toxic and have to be eliminated without complications by the body in a short amount of time (<24 hours).
- The presence of the contrast agent must favour the visualization of the area of interest relative to the surrounding fluids and tissues by, at least, a factor of 2.
- Reaching the target area by the ionizing element must be sufficient to perform a complete CT scan inside the hospital.
- The contrast element have to be readily soluble or form stable suspensions at aqueous physiological conditions.

Obviously this should also be done to reduce the amount of radiation that affects the patient during the medical examination.
There are different contrast media that can be used, which are able to change the amount of X-rays through tissues. For this reason they can be divided according to the fact that they have a higher or lower absorption than the target organ: there are positive contrast agents, like barium and iodine, and negative, like air and carbon dioxide. The positives have a elevate probability of interacting with the emitted photons because they have a high atomic number (56 for barium and 53 for iodine). The first in particular, present as Barium sulphate, is used for digestive tract examinations. It is given by oral via; it remains in the intestinal lumen without being absorbed or causing toxic reactions and then it is eliminated through the stool.
On the other hand, the use of iodine alone has slowly been replaced for medical applications due to its high concentration and toxicity. Therefore, the use of contrast media with iodine atoms bound to organic molecules, has become widespread. Initially, many ionic contrast agents with iodine were studied, which showed a strong tendency to interact with biological cells, like peptides and cell membranes. In addition, it was observed that such instruments with a high osmolality (8 times higher than that of plasma) caused a high possibility of adverse reactions in the

body and increased pain and heat sensation in the area where they were injected. This led to the development of a more recent category, the non-ionic iso-osmolar contrast agents, which had an osmolality equal to that of blood plasma [2]. These, however, don't present a lower quantity of iodine inside but they show the same contrasting effect inside the organs. At the moment, they are used especially for CT exams, because the speed of new devices and the diffusion of automatic injection instruments, have guaranteed the control of the quantity and the speed of injection [7]. In this way the scans execution times is optimized.
Despite the introduction of non-ionic contrast media has almost completely eliminated ionic contrast media from the market, the incidence of side effects has not completely disappeared, although it has considerably decreased. In fact, the presence of side effects is around the 3% of patients.
Adverse reactions can be divided into two categories: the predictable and the unpredictable [2].
The first ones are linked to pre-existing pathologies within the patient, and depend on the quantity of the contrast agent inserted. For this reason they are not always used for fear of excessively negative reactions.
The second category is linked to allergic reactions of the patient, caused by molecular elements within the contrast agent, and they are independent of the dose administered. The kidney in particular is the organ most subject to toxic reactions, leading to the formation of acute renal failure, or a deterioration of an already existing renal failure. The administration of contrast medium is also not recommended in all patients with diabetes mellitus, heart failure, dehydration or who have already taken nephrotoxic drugs.
In conclusion, in these situations, the use of contrast agents is strongly not recommended or even forbidden. In this way, it is impossible to obtain medical images of higher quality, useful for diagnostic purposes. All these situations limited the possibility of obtaining new data.

1.3 Abdominal CT

Regarding the analysis of the abdominal component, CT has recently established itself as a second instance method, bypassing traditional radiology, demonstrating that it is very useful in the management of patients with acute abdominal pain [8]. This imaging technique, in fact, allows a good visualization of arterial and venous vessels, parenchymes and main abdominal organs. Furthermore, one of the main advantages is that it allows the direct visualization of all the components of the urinary apparatus, that are external to the excretory system. However, image variations caused by the presence of contrast are not necessarily uniform and depend on numerous physical and biological factors, like blood flow in tissues or extracellular component in a biological constituent, or even on the physical characteristics of the machinery itself.

The development of special equipment, such as spiral and multi-layer CT, has given the possibility of a three-dimensional examination of the vessels and, at the same time, a less invasive diagnostic examination (like Angio-CT). The injection of the contrast agent into the blood vessels takes place with a very rapid injection rate, approximately 3 ml/sec, using an automatic injector, after supplying the patient with at least 500 ml of water [9].
It is possible to identify 4 different phases, depending on the time between contrast administration and patient irradiation. The arterial phase takes place first, approximately after 15-25 seconds, where the renal arteries are displayed as hyperdense; while the renal veins are displayed at the end of the arterial phase. After approximately 30-40 seconds, it is possible find the corticomidollar phase, which is contradicted by an improvement of the renal cortex. This is followed by the nephrographic phase (after about 80-120 seconds), where there is an opacification of the renal parenchyma. Finally, after a few minutes of contrast medium insertion, there is the excretory phase. Which is characterized by an opacification of the ureter, followed by the elimination of the contrast agent from the organism. It is obviously possible to make acquisitions of multiple phases if necessary, for diagnostic purposes.
In the Figure 1.2 it is possible to observe how two CT axial images appear before and after the insertion of the contrast. In particular, thanks to the use of contrast, it was possible to identify the cystic lesion in the left kidney [10].
Therefore, the first image, without the insertion of contrast, shows the difficulty from a diagnostic and an image processing point of view, to identify the renal component from the other abdominal organs. This is caused by a greater homogeneity of the data on the image. The use of contrast facilitates the identification of these structures. In conclusion, the generation of this type of image is extremely useful for diagnostic purposes and for image processing models.

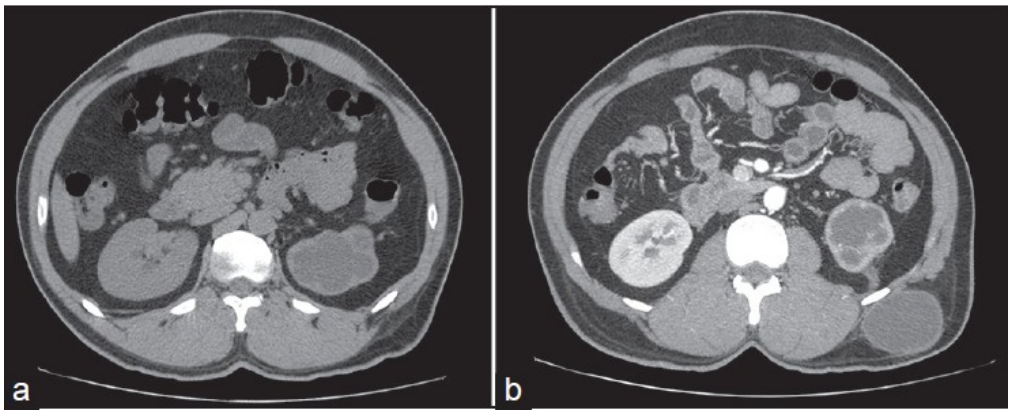


Figure 1.2: (a) Non-contrast axial CT of the abdomen. (b) Contrast-enhanced CT (with arterial phase), axial section showing a cystic lesion located in the subcutaneous plane. Adapted from [10].

Chapter 2

Learning Deep Generative Models

As briefly described earlier, the core of the project carried out at Telecom Paris is based on the study and application of a particular generative model, which is able to produce the required data.
These particular models have spread especially in recent years, with an increasing number of studies every year and is applied to numerous sectors, from music to image formation. It can be briefly described as: a generative model describes how a dataset is generated, in terms of a probabilistic model. By sampling from this model, we are able to generate new data [11]. To underline its main characteristics, it is necessary to compare it to another model, the discriminative modeling. The main difference is that every observation in the training set has a label (for this reason it is often associated to supervised learning, where it learns a function that is able to map from an input to an output with the help of labeled training set). In this way the model estimates $p(y|x)$, that is the probability that the sample x belongs to class y. The real class to which input belongs is called "desired output", while the error of the network can be easily identified as the difference between the desired output and the real output of the classifier. Even a generative model can be trained through a labeled dataset set, but not necessarily; its main goal is to understand the probabilistic distribution of different samples.
The model also includes a random element that is able to generate new elements, but with the same features. Therefore, this characteristic allows one to work with numerous datasets, whose label is not always applicable.
To fully understand these concepts, it is necessary to clarify the elements present at the base; starting from simple neural networks models up to the most complex deep convolutional neural networks.

2.1 Unsupervised learning

We speak of unsupervised learning when the data for training consists of a set of input vectors without any corresponding target value. In fact, the model performs a sort of cluster, reorganizing the inputs and identifying a series of common characteristics to try to make predictions and considerations.
This leads to a greater difficulty to understand the learning process of the model, because we do not know exactly on which input characteristics it is based. In the study described in the following chapters, this type of learning is largely protagonist.

2.2 Neural Networks

In machine learning, systems able to learn with dataset. These are also evaluated through a set of data called test sets, which are usually different from the data used to train my system to identify a logical relationship within them. Neural networks (NNs) is one of the most common structures of machine learning, whose name depends on its close similarity to the biological structures of the nervous system [12]. The Figure 2.1 summarizes the main elements that compose a NN.

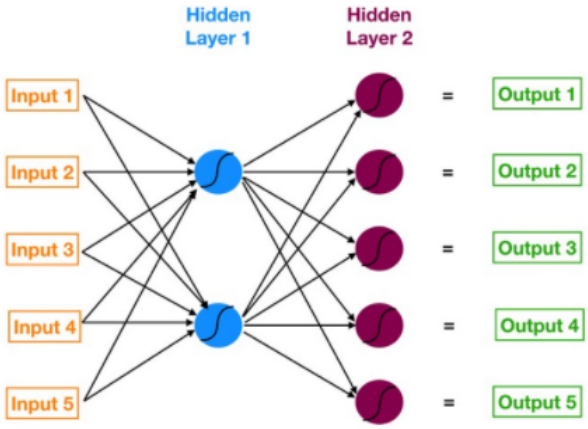


Figure 2.1: An example of a NN structure.Adapted from [12]

On the left there is the input layer, where the network receives data from the outside (made up of 5 elements). Subsequently the information is propagated through two hidden layers of neurons and, at the end, the information passes to the output layer which represents the prediction of my model.The number of hidden layers can be of different values, depending on the model you want to develop. The arrows that connect the different dots show how neurons are connected to each other and

how information travels through the network. These connections are characterised by weights, whose value contains the key information that the network is able to process.
The weights are initially initialized with random or predefined values and then, depending on the analyzed data, change their value to facilitate the learning process. An epoch is when all the data has been passed through the network; then the data is passed again with the weights now changed, in this way the NN "learns".
The information travels from the input layer to the output layer, and is known as forward propagation. When it reaches the output layer, its quality is evaluated in order to train the whole network.

2.3 Loss Function

The network output does not match the target output, the difference is used to calculate the loss function (L), so you can see "how well my algorithm is working". The network has to modify its own weights (and the relative bias), to try to minimize L. There are numerous functions available, so it is essential that the chosen function is able to capture the characteristics of our problem, and be motivated by concerns that are important to the project [13].
There are different types of functions depending on the model, two main categories can be distinguished, and they are associated with regression or classification problems. The project is related to the first group. In the study it was decided to use the functions described below.

2.3.1 Mean Square Error (L2)

$$MSE = \frac{\sum_{i=1}^{n} |y_i - \hat{y}_i|^2}{n} \tag{2.1}$$

MSE is the most used function in case of regression problems. Where y_i indicates the output of the model, $\hat{y}_i$ represents the desired output and n the number of elements. This indicates the sum of the squares of the distances between the target and the predicted values. L2 has a higher gradient for large loss and decreases when it approaches zero, making it more accurate at the end of the training phase, where a lower loss is expected.

2.3.2 Mean Absolute Error (L1)

$$MAE = \frac{\sum_{i=1}^{n} |y_i - \hat{y}_i|}{n} \tag{2.2}$$

Where y_i indicates the output of the model and $\hat{y}_i$ represents the desired output and n the number of elements. In this case the regression metric is based on the average

of errors within a group of predictive values, without considering their direction. This is the average of the modules of the differences of the predictive values. This function is also more stable if there are outliers within the training data.

2.4 Deep Learning

Deep Learning (DL) is a subcategory of machine learning, where you find models such as neural networks. DL was developed as early as the 1980s. It is currently one of the main fields of interest related to AI technology. It was spread at the beginning of the new millennium thanks to the arrival of big data, which favoured economic and scientific interest in this sector.
In addition, these models are now known thanks to improved hardware structures, computational power (GPU) and the availability of large datasets. The NNs used in this field exploit a high number of hidden layers in order to obtain a higher level of abstractions, in this way very complex problems can be solved through features recognition [14]. The Figure 2.2 below summarizes the main elements present within a DL model.
One of the specific frameworks for DL is tensorflow, which provides optimized modules for algorithm development, but the entire project was written thanks to Pytorch, an open source project developed by Facebook. A framework widespread especially in recent years, which uses a technique known as dynamic computing to simplify the training of NNs, and the different commands imitate the conventional programming model, making it very popular especially for researchers and engineers.

2.4.1 Activation Functions

The activation function f determines the output value Y of the neuron, based on the sum of the input values, which are multiplied by weights(w). This relation $Y = f(\sum w * x)$ describes the situation.
There are several functions which are characterized by some properties, like the range. When the range of the function is limited, the training tends to be more stable because pattern presentations significantly affect only limited weights. While an infinite range causes a more efficient training, but a smaller learning rate is needed because the weights vary more.
Now the activation functions that have been considered in the project are briefly described.

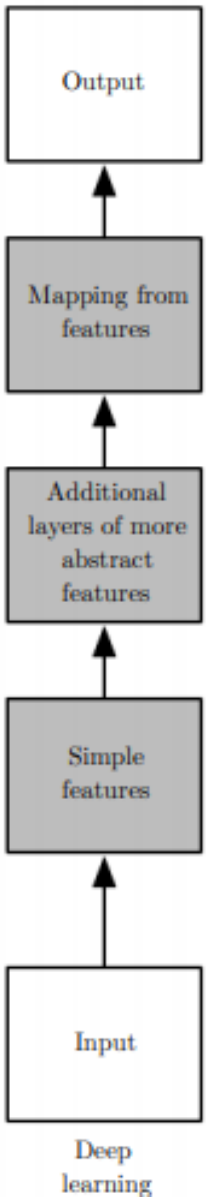


Figure 2.2: Flowchart of a DL algorithm. The grey boxes indicate the component that are able to learn from data. From [14].

2.4.2 Rectified Linear Units (ReLU)

This particular activation function, as you can seen from the graph, returns zero if the input is less than 0, instead if the input is greater than 0, the output is equal to the input. Therefore the output range has only a lower limit.
ReLU is a non-linear function and has the advantage, compared to other popular activation functions, of speeding up the model. Its application spread especially in

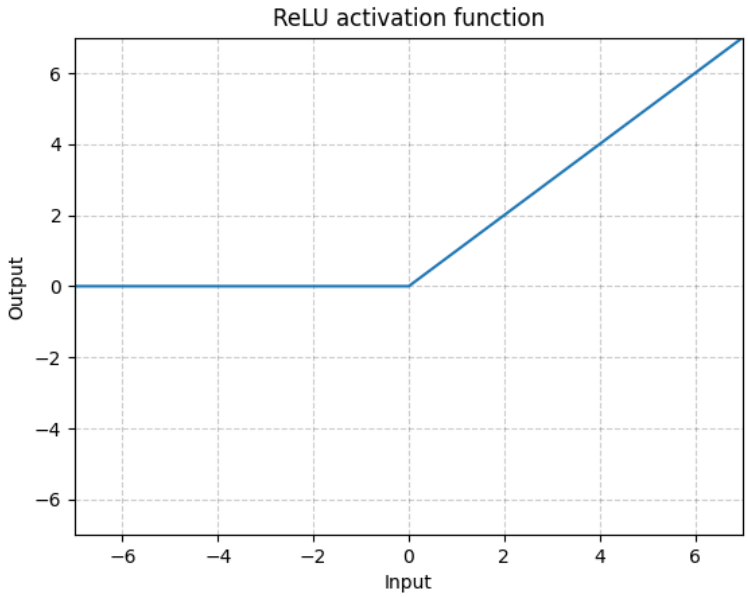


Figure 2.3: The graph of ReLU function. From the documentation of Pytorch

Deep learning applications, and other slightly different activation functions soon became popular. In fact, one of the main limits is that the neuron remains completely inactive when input is less than zero, significantly reducing the flow of information when the number of "dead" neurons is very high.
Leaky ReLU slightly is different because on the left side of the graph, the output is not completely null but has a slight slope. This means a small leak, and avoids the "death" of the neuron in the network by extending its range.

2.4.3 Hyperbolic Tangent Function

Moreover, as recommended by several studies, the Hyperbolic Tangent Function was also used in the models of the project. This activation function is very similar to the classic Sigmund function, but in this case its codomain goes from -1 to +1 (Figure 2.4) .
As described in the following chapters, this function is fundamental to guarantee stability during the training process of our generative model

2.4.4 The optimization algorithms

In the NNs there are different optimization models, which iteratively minimize (or maximize) an objective function ($J(\theta)$). The classical gradient descent method updates the parameter θ along the opposite direction of the gradient of the function

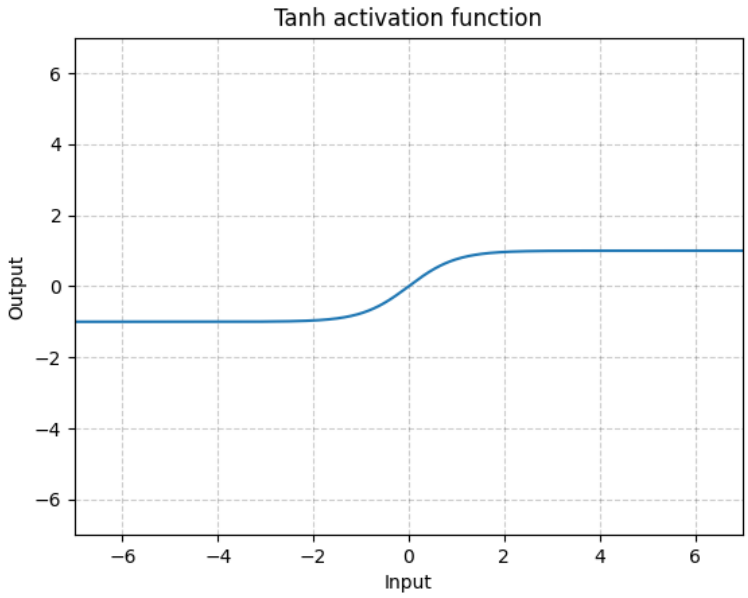


Figure 2.4: The graph of Tanh function. From the documentation of Pytorch

(J(θ)), where the learning rate η determines the width of the variation.

$$\theta_{t+1} = \theta_t - \eta \nabla_\theta J(\theta) \tag{2.3}$$

However, for the computation of ∇(J(θ)), it is necessary to use all samples in the training set, risking to converge to a local minimum or an excessive variance.
For this reason other models have been developed. These models update the gradient after a specified number n of samples (mini-batch), to make convergence more stable and reduce the variance of the gradient. Or, alternatively, the stochastic gradient descent method (SGD), updates when a random sample is taken at each iteration.
The introduction of the momentum was a big innovation, to speed up the learning process. The idea is to calculate an exponential moving average of the previous gradients, to optimally direct the update of θ.
The optimizer ADAM [15] (Adaptive momentum estimation) is based on two coefficients. ADAM memorizes the exponential moving average of the squares of the gradients and the exponential moving average of the previous gradients. In this way, it exploits the first moment, the mean (m_t), and the second, the variance (α_t), of the gradient. Where η is the step size (it can depend on iteration). This method to optimize the gradient is the most efficient to stabilize the CycleGAN during the training phase. In particular it was demonstrated that reducing the momentum $\beta 1$ from 0.9 to 0.5 helps stabilize training [16].

The (2.4) indicates the optimizer ADAM:

$$\theta_{t+1} = \theta_t - \eta \frac{m_t}{\sqrt{\alpha_t} + \epsilon} \tag{2.4}$$

2.4.5 Polling Layers

One of the problems related to convolution is the creation of future maps that are too sensitive to the position of the different features. In addition, there is also the problem of overfitting, i.e. when the network is so tuned with data during the training phase that it not able to generalized for validation and test sets. In these cases you may notice a drop in performance during the test phase compared to the training and validation phase.
Pooling Layers down-sample the features maps, providing benefits to the whole structure and reducing computational costs. Even if the image has a lower resolution than the input, the main structures remain. This process also causes a denoising on the images.This procedure guarantee to keep the representation approximately invariant to the small translations of the input [14].
There are two main common pooling operations:

- Max-Pooling: where you take the maximum value between pixels for each patch in the feature map.

- Average-Pooling: where the final result is the average between the pixels within the patch

The first technique works better, because in a nutshell it is more informative to analyse the maximum presence of different features rather than their average [17]. Therefore, sometimes the most efficient strategy for down-sample and avoid information loss is not using a Stride Value greater than 1, but selecting the main feature activation on small patches.
However, it has been observed that in GANs it is more useful to use a larger Stride rather than max-polling for the generator, allowing it to learn its own spatial upsampling [16].
The Figure 2.5 shows the result of the application of this technique.
Among other techniques to avoid overfitting, there is Dropout which consists of randomly setting a certain number of neurons to zero (Dropout Rate) [18]. This strategy was also applied in the model; the generator in this way, tries not to focus only on some features.
The choice of this value as well as many other hyper parameters, among which the learning rate has a great relevance, favour the gradient descent to reach the solution of the problem.

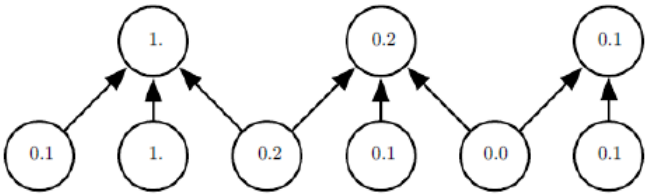


Figure 2.5: *Pooling and downsampling.* The figure shows a max-pool width of three and a stride between pools of two. This application reduce the number of feature. Reprinted from [14]

2.4.6 The Batch Normalization

The training phase of a NN is very sensitive to the different parameters assigned and also to the values of the weights at the beginning of the first epoch (this will also be specified later for CycleGAN).
The Internal covariate shift is a problem of instability during the training phase, this is linked to the fact that all layers are updated from the output to the input. The training of the nets is conditioned by the fact that the input distribution of each layer changes, due to the change of the previous one. This leads to a slowing down of the training phase and the achievement of convergence [19].
Batch Normalization is a method realized to coordinate the updating of the different layers.It re-parameterize the model so that some units always remain standardized [17], stabilizing the entire process during the update of the layers.
Batch Normalization is based on standardization by calculating the average and standard deviation of each input variable for each mini-batch.

2.5 U-Net

Now two types of deep networks are presented, which have been very popular recently:the U-Net and the Res-Net. These models are used to build the two generators in the Generative Adversarial Network, because they showed a great ability to identify the main features on which the generative model must be based in order to create new data. The U-Net is studied to exploit data augmentation in order to work with datasets that are not too large; especially in the biomedical field, where it tries to gain an output that includes localization (a label is assigned to each pixel).
The U-Net is a fully CNN, and its name derived from its particular architecture, it was designed for the segmentation of biomedical images [20]. As can be seen from the Figure 2.6, the left-side consists of a contraction path: composed of three 3x3 convolutions, each followed by the respective activation function (ReLU) and a 2x2 max-pooling for downsampling. In the latter case the number of feature channels

is doubled each time, to allow the network to propagate information up to the higher resolution layers. In the right side there is the expansion path, where 2x2 convolution (called up-convolution) are applied to halve the number of channels and upsampling. There is also a concatenation with the cropped feature map with the same size, obtained directly from the left branch of the network. In this way, the localisation information is preserved. Also in the right-side ReLU are applied. In the last layer, a 1x1 convolution is applied to analyse each feature vector, depending on the number of desired classes.
From this network architecture, several modifications were applied for biomedical imaging: such as the 3D U-Net for volumetric segmentation [21] or the V-net, where the segmentation of MRI prostate volumes was optimized thanks to the Dice overlap coefficient, between the ground truth and predicted result [22].

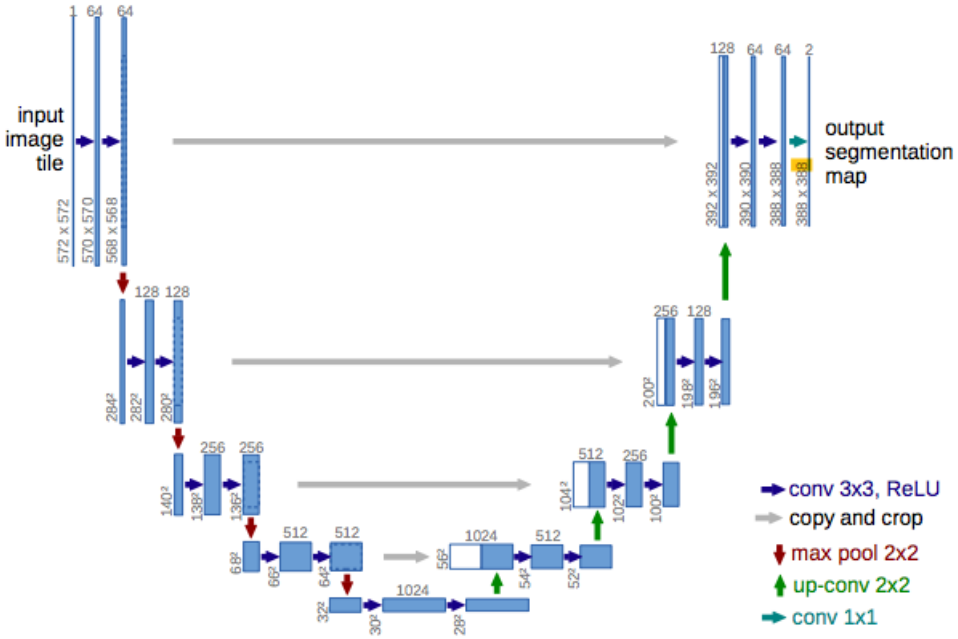


Figure 2.6: The architecture of the U-Net. Reprinted from [20]

2.6 Residual Networks

The deep Residual Network (ResNet) is another model of CNN that has been very popular and used, since it was first introduced in 2015 [23].
This structure was used to create the generator within the CycleGAN, as will be shown later, the ResNet demonstrated greater stability than the U-Net during the training phase.

Before its introduction, the NNs, with increasing depth due to the number of layers, were characterized by a saturation or even degradation of the performance. Furthermore, ResNet solves the notorious vanishing gradient problem, i.e. when the gradient is back-propagated up to the first few layers, reducing until it disappears completely. To solve this problem someone try to add, for example, an extra loss as supervision, but it was useless [24].
The winning idea of this model is to add an "identity shortcut connection", to skip one or more layers [23]. This does not degrade the accuracy of the network, because the NN stack simply identity mappings and performance are the same. The fundamental element is the "block", consisting of a series of identity mappings where, at the end, the input of the block is added to the output (Figure 2.7). This operation of "addition" is performed element by element. In the case where the input and output have a difference size, different padding techniques can be applied to concatenate the elements. The central idea is that letting the stacked layers fit a residual mapping is easier than letting them directly fit the desired underlaying mapping.
This model has made it possible to develop NNs with an increasing number of layers, avoiding, at the same time, degradation of performance and gain accuracy from a considerably increased depth.

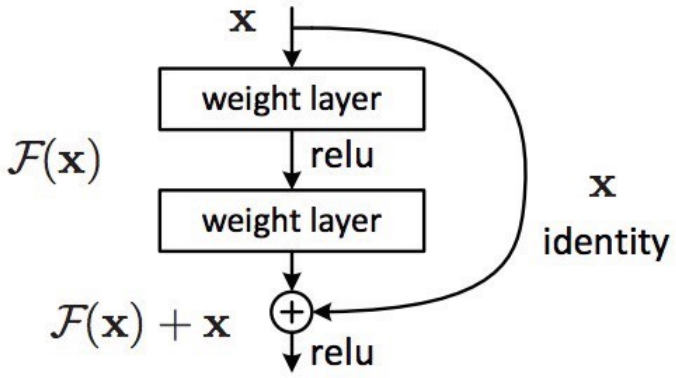


Figure 2.7: Residual learning: a building block. Reprinted from [23]

2.7 A Deep Generative Model: the GAN

The GAN is a generative model, consisting of two fundamental elements: the generator (G) and the discriminator (D) [25]. They are composed by deep NNs to produce data from a known distribution. The input of G is z, which is pure noise. The output must be as similar as possible has a random sample chosen by the data distribution. The input of the discriminator instead is a vector randomly chosen

from the samples synthesized by G or from the real distribution, its output indicates the probability that this element has been created by the network generator (Figure 2.8).

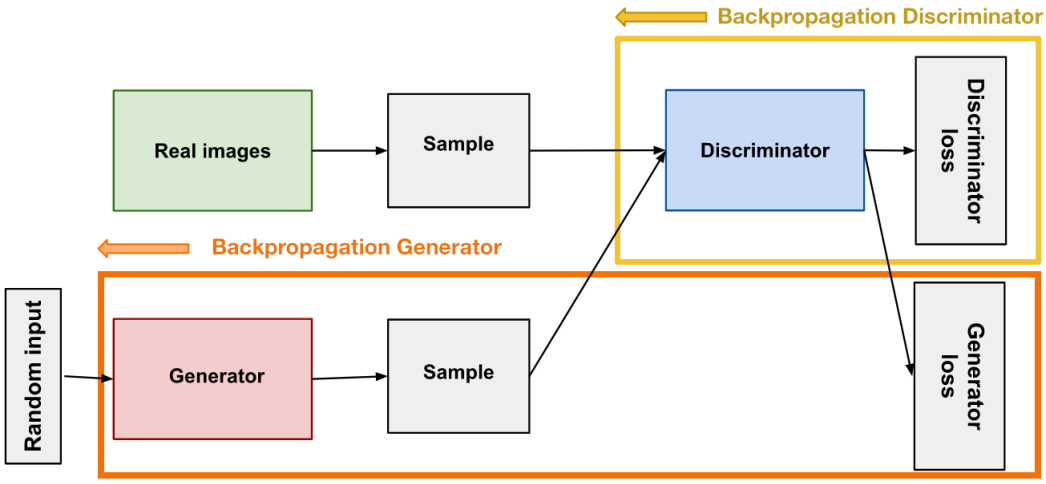


Figure 2.8: Model of a GAN

The main objective of the generator is to be able, not only to create data as realistic as possible, but also to be able to fool D, this instead tries to identify as much as possible the fake data from the original ones. In this first and simple description of a GAN, the information is back propagated from D to G, in this way the generator adapts the weights of its network only thanks to the information obtained "from his competitor". Training can be seen as a mini-max game between two adversaries. The objective functions of both networks can be summarized with the equation:

$$E_{\mathrm{x}} \log(D(x)) + E_{\mathrm{z}}[\log(1 - D(G(z)))] \tag{2.5}$$

The generator tries to minimize it and the discriminator tries to maximize it. The G(z) represent the output of G when given noise z, while D(x) is the probability that the data is real.
Often both networks are optimized with the same step, but in different models, it is necessary to alternate through k steps the optimization of the discriminator with one step instead for the G. In this way we try to avoid a too weak discriminator, which would lead to the generation of images whose data distribution would be too different from the original. On the other hand, it is also necessary to avoid a classifier that is too strong and a generator that is therefore unable to fool it.
It is observed how D tries to maximise the log-likelihood to estimate the probability that, given x, this belongs to the real distribution, while the generator tries to do the opposite thing.

The discriminator often presents a common architecture, composed of several convolutional layers and a dense output, which contains the probability of the input to belong to it. However, in the original project, the discriminator was composed of dense layers, which led to a lower predictive ability. Furthermore, several variants regarding loss were analysed to improve the abilities of the discriminator through time, such as the use of class labels combined with the use of cross entropy loss [26]. Moreover, to stabilize the discriminator during the training phase, several losses have been suggested over time: starting with vanilla GAN [25] based on Binary Cross Entropy, this function is used between the output vector and the response vector composed of 1 or 0 depending on the corresponding label. Other loss functions were based on Wassertein distance [27] (obtaining the models known as WGAN) or f-divergence [28]. These variations were often adapted to the type of data that the GAN need to train on, to ensure higher stabilization and avoid collapse mode. Collapse mode happens when the generator focuses only on some features or samples, which are able to fool the discriminator without adding more information. Basing these observations only on, the loss gradients precipitate very quickly towards zero with the conviction that they are achieved to convergence, even though this is linked only to a few significant samples of the input distribution. Obtaining, in the end, only a weak generator that is not able to improve its output vector.
The use of different losses was also determined by the common objective of reducing another major problem linked to the training of GANs, "the oscillating loss". A great oscillating loss can often produce non-optimal results, making the convergence of the model impossible. The ideal would be to obtain a stable loss, with a constant increase or decrease. Moreover, freezing the discriminator weights is fundamental to stabilize it during the training of the generator, where its weights will be updated in relation to the chosen loss value. All this is necessary to make sure that the discriminator is not too able to identify the images generated, preventing the optimization of the generator.
On the other hand, for the generator there is often a more complex architecture. It has the same goal as a decoder of a Variational Autoencoder, which is to convert a vector into the latent space of an image, whose distribution, in this way, allows new data to be generated.
Also, for the generator, there have been several transformations of its original architecture; an example is the use of a variational autoencoder network (VAE), where we can find the pixel-wise reconstruction loss to generate new data from the original [29].
Another very common type of structure for generating new images is conditional GAN (CGAN), so called because extra features or conditions (like class label, text descriptions, object locations or sketches) related to the desired properties in the output, were added to the discriminator to obtain higher quality results, during the generation [30]. In this way the model forces the difficult relationship between the

latent space of the generator and the output of the new images.
Often image pairs, have been used as input through different supervision models. For example, based on segmentation with dice loss [22]. The dice loss allows to measure the overlap between two sets, related to the foreground and background of the image.

2.8 The CycleGAN

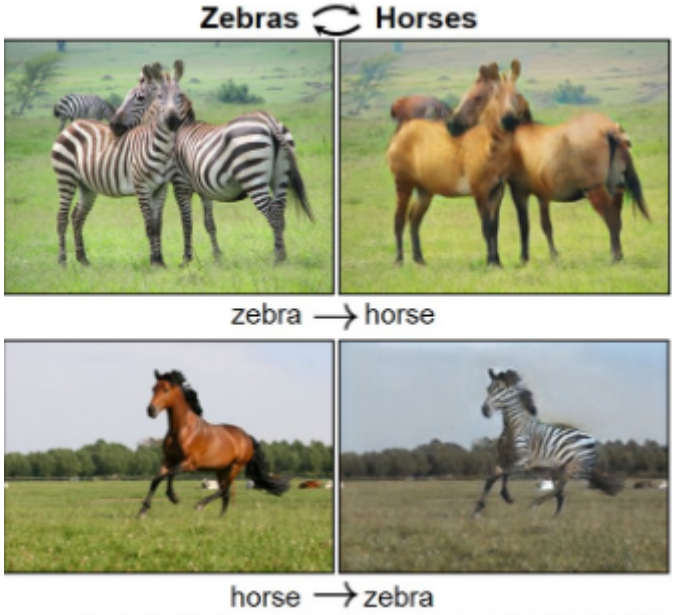


Figure 2.9: Examples of CycleGAN application using zebras and horses from ImageNet. Reprinted from [1]

The CycleGAN represents a different generative model, mainly based on the integration of two different GANs. The codes and implementation of the model was on Pytorch. It was mostly based on the recent work of Zhu et al. [1], where the CyleGAN was described and examined.
The principal characteristic of this network is its ability to generate new data, overcoming the limitation of working with a dataset of paired images. This is necessary for others popular transfer models, which allows the translation of images into different domains, like pix2pix [31].
Therefore CycleGAN, is significantly more adapted to work in medical field because it is rare to have pairs of images of the same patient, acquired with different techniques or characteristics.
Thanks also to the wide area of application, another model called UNIT [32] specific for unpaired samples, has been created. This generative model, unlike CycleGANs,

is composed of two variational autoencoder networks (VAESGANs), which both work on two different domains, but they share the same latent space. In different papers these models and their performances are often compared due to their ability to work with the same type of data.
Thanks to its particular architecture CycleGAN is able to train image translations at the same time from both domains, from X to Y and vice versa (Figure 2.10). It is composed of: a generator G_{xy} to transform CT images from Y to X (indicates as F), and G_{yx} to translate the image from X to Y domain (indicates as G).
There are also two discriminators (D_a and D_b) to improve the quality of the images, identifying the real images from those synthesised by the generators. The work done by the discriminators determinate the adversarial loss to to update the weight of the NNs. In addition to this there are two other contributions from: Cycle Consistency Loss and Identity loss.

2.8.1 The Cycle Consistency Loss

The goal of the two generators (G and F) is to create a mapping function that is able of generating data corresponding to the target distribution. In addition they should have the ability to always learn from the same dataset to generate random permutations in the target images, where the learned functions are still able to create a proper output distribution. For this purpose the adversarial loss alone is not enough, it is necessary to introduce the Cycle Consistency Loss.
The latter is based on taking a sample ($\hat{x}$) generated by G_{xy} and using it as input for the G_{yx} generator. The same thing is done for the other domain. In this way the model tries to reconstruct the starting image, for both generators.
The Equation (2.6) represents the loss:

$$\mathcal{L}_{cyc}(G,F) = \mathbb{E}_{x\sim p_{data}(x)}\left[||F(G(x)) - x||_1\right] + \mathbb{E}_{y\sim p_{data}(y)}\left[||G(F(y)) - y||_1\right] \quad (2.6)$$

The loss of the model in this case is evaluated through the L1 norm.
In the fourth chapter, as suggested in [33], its influence will be reduced in the last epochs, during the training phase, because it can induce the generator to not evaluate the necessary changes to improve the quality of the output images.
The Figure 2.10 shows the scheme of the forward/backward Cycle Consistency Loss, this technique is already known in the literary environment, such as the field of literary translations, called "reverse translation and reconciliation" (M. Twain mentions it in his works).
However, the application of Cycle Loss alone is not enough to generate realistic images, indeed it has been observed that it alone often causes mode collapse, producing images that are perfectly identical to the input images.

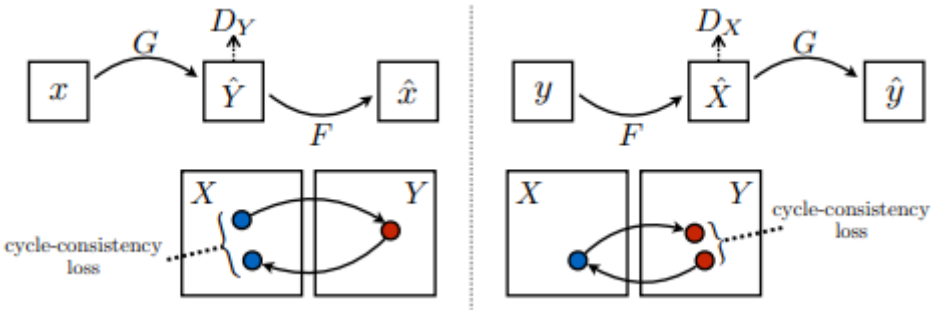


Figure 2.10: Diagram of how cycle loss works: on the left, there is the forward Cycle-Consistency, on the right there is the backward Cycle-Consistency. Reprinted from [1]

2.8.2 The Identity Loss

Identity loss is a supplementary parameter that is added to improve the quality of the generated images. It is added in order to preserve and improve the colour in the images, which is extremely useful in CT images with contrast, where the intensity and variation of the pixels is essential for diagnostic purposes.
It is based on taking as input a sample x, belonging to the target domain X, and passing it through the G_{xy}, in this way the generator learns an identity mapping function.
The L1 norm is used to evaluate the loss [1]. The Equation (2.7) represents the loss:

$$\mathcal{L}_{identity}(G,F) = \mathbb{E}_{y\sim p_{data}(y)}\left[||G(y)-y||_1\right] + \mathbb{E}_{x\sim p_{data}(x)}\left[||F(x)-x||_1\right] \qquad (2.7)$$

In the Figure 2.11 it is possible see how the application of this element to improve image quality.

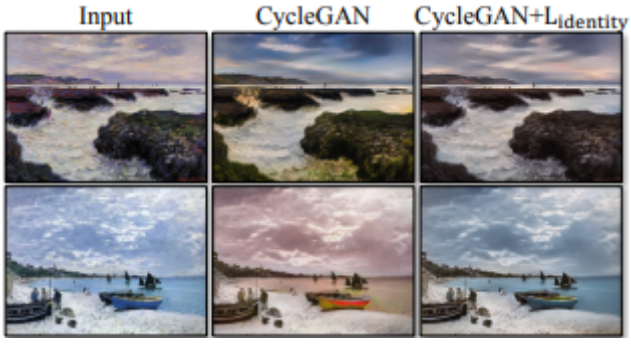


Figure 2.11: Examples of the effect of Identity loss on Monet's paintings. Reprinted from [1]

2.8.3 The architecture of the Discriminators

The two discriminators inside the model must return the probability that the image furnished belongs to the real distribution or is a fake.
There are several tips that help in this objective.
Actually, the discriminator does not assign the probability to each pixel of the input image, but often divides it into overlapping zones called patches, and assigns the probability to each of them according to the content of the different features. This technique is called PatchGAN. In this way, the performance of the discriminator is not completely based on the content of the input, but it is focuses on its features.
In the realized model (described in the method proposed), the discriminator is composed of 3 convolutional layers. Where, starting from an image with one channel, at the beginning a filter with 64 channels is applied and at the end 256 features channels are obtained, the stride is 2. To avoid the vanishing gradient problem, by zeroing a large number of neurons in the NN due to the ReLU, the activation function Leaky ReLU is used with a gradient of 0.2 [16].
To initialise the weights, a normal distribution with mean equal to zero and standard deviation of 0.02 was used. All these characteristics are described in [1], they are applied in the CyleGAN model developed, reported in the fourth chapter.
The aim of the discriminator is opposite to that of the generator, for this reason sometimes its loss grows in an exponential way (indicating that it is too good to recognise the real images), in this case it is necessary to use as input noise images to fool it. Another alternative, applied during the project also, is to invert, with a probability of 5%, the labels relative to fake and real images, to avoid training a too weak generator [33].
Another suggestion, to prevent an excessive oscillation during the training phase of the model,is using as input of the discriminator not the last synthesized image, but an image chosen randomly from a set of K elements [34]. This also avoids the problem called "absence of memory", which can cause divergence if the discriminator does not notice some artifacts. In this way, the generator, will apply them again for all future images. A buffer of K images is created, and a mini-batch of k images. At each iteration the discriminator takes $k/2$ images, which are then replaced by others generated in the next iteration. In this way the buffer size (K) is constant. This particular technique is included in the CyleGAN model developed, with a K equal to 50.

2.8.4 The architecture of the Generators

Two models of DNNs were used for the generator architectures: the U-Net and the ResNet.
For the latter the construction was based on the model of Johnson et al. [15], where 6 blocks are used to train images with dimension 128 x 128. In the case of 256 x 256 images , the number of residual blocks is 9.

In the first layer there is a convolutional layer, with a kernel 7 x 7, where 64 filters are applied and with a stride of 1. The activation function is ReLU. The following layers are characterised by a kernel with dimension 3 x 3 and with a stride of 2. In addition, the technique of reflection padding was used to keep the size constant after each convolution.
While for the U-net, there are 64 filters in the first layer, and the number of down-samplings is 7. For this reason an image 128 x 128 arrives at the bottleneck with dimension 1 x 1. For the convolutions in the decoder section, it was applied a 2D transposed convolution operator over the image, to reduce the formation of artifacts in the image synthesis [16], with a 4 x 4 kernel, a stride 2 and a padding 1.
For all layers the batch normalization is applied, to reduce the stability of the network and its oscillations. It is applied to both the discriminator and the generator, except for the output layer of the generator and the input layer of the discriminator.
For both the models of DNNs, in the output layer the hyperbolic tangent (Tanh) activation function is applied, int his way the data are scaled between [-1, 1]. For the remaining layers the ReLU activation function is applied to promote sparse activation.
To conclude, as suggested by several papers, dropout is applied to improve the sparsity of information, putting some weights to zero and to reduce features dependencies in the network (reducing the problem of overfitting).
The architecture of the ResNet and the U-net described in [14], were also applied for the NNs which are used to compose the CycleGAN model of this study, described in the fourth chapter.

Chapter 3

The State Of The Art

The lack of a large quantity of data is one of the main problems related to medical imaging, this is due to several factors from the high cost of production to problems related, to the privacy of patients. All these factors represent a significant problem from the point of view of deep learning, because sometimes the lack of data leads to a worse performance of the networks.
Therefore, generative models, have been extremely well-treated in recent years, as they represent an excellent way of overcoming this gap, trying to produce new data. These are very difficult to assess quantitatively. At the moment, in fact, the images generated are evaluated mainly by visual inspection. However, in the results of the project, a new method for evaluating images is presented, which has been increasingly used recently.
From the various studies presented, the backbone is the paper introduced by Goodfellow [25], who presented a new network model, called Generative Adversarial Network (called GANs for short) and pointed out that one of the main future challenges of this new model involved exactly the application in the medical field and the opportunity to solve the main challenges in this sector.
This chapter will present the main evolution and the use of the Generative Adversarial Networks, underlining how their performance and application have evolved over the years to the present day, creating progressively more complex structures such as the Cycle Consistency Adversarial Network.
In the last few years several studies have been carried out in different applications in the medical imaging, these areas can be divided into: reconstruction, image synthesis, segmentation, classification, detention and registration [35]. In particular, this chapter makes a review of the state of the art of GAN's application in the medical sector, more specifically in image-to-image-translation.

3.1 Applications of GAN to generate medical images

The GAN is a particular generative model characterised by the simultaneous training of two different networks: a generator (G) and a discriminator (D), sometimes called the critic. The term adversarial is given by the fact that both nets have the opposite goal.
The main objective of the generator is to be able, not only to create data as realistic as possible, but also to be able to fool D, which instead tries to identify as much as possible the fake data from the original ones.
The first element of the network, the generator, has the job of learning the distribution of data from a specific dataset, in this way it can generate new data with as similar a distribution as possible. The discriminator is its counterpart, whose purpose is to identify and distinguish between the images generated by the generator from original images, taken from dataset. For this reason, this network can be described simply as "a game between two opponents", the generator and the discriminator. Its main features were discussed in the previous chapter.
The generation of new images, starting from simple noise or pre-existing medical images, is also studied to improve the performance of Deep Neural Networks during the training phase. It provides more elements to train, compared to the classic data augmentation where there are: transformations, rotations or zooms. However, these simple variations in shape, size or contrast can sometimes lead to performance deterioration for a classifier or segmentor network.
Recent studies have tried to exploit GANs, for the unconditional generation (i.e. without the addition of special conditions to facilitate the generator's job). There are several variants of the simple GAN for this purpose: such as the Deep convolutional Generative Adversarial Network (DG-GAN) or the Progressive Growing of GAN (PGCGAN). In the latter, in particular, the main characteristic is the growth of both the generator and the discriminator, starting from a low resolution and then adding layers that increase fine detail.
In [36] a DG-GAN was used to generate images of lungs with benign or malignant nodules, to try to analyse the main features present in both cases, to facilitate the identification of any pathologies by radiologists. The study was carried out on 2D images, and the evaluation was just based on a visual analysis by two professional radiologists to assess the quality of the fake nodules present. Possible future applications would have involved the training of possible networks or the training of future doctors. In addition, the images were subjected to an intensive screening and selection process. This limits the amount of data available. However, in this project, efforts were made to generalise the type of images used as input as much as possible. The model is also limited to 2D data only.
While in [37] the generator is trained to translate from simple noise images to high quality images containing liver lesions. These images were then used to train

a CNN, comparing the results with the classic data augmentation. The latter method, in fact, caused excessive image variations, reducing in this way network performance. The performance of the classifier increased from 78.6 % sensitivity and 88.4% specificity to 85.7% sensitivity and 92.4% specificity, thanks to fake images.
In [38] a PCGAN is used to translate low quality images into mammograms with high resolution and highly realistic. To realize this project, a large dataset with over one million images was used, but excluding images containing post-operative artifacts or presenting extraneous elements to the body (like metal clips, pacemakers or stents). Despite the instability of the generator, caused by the fact that the optimal point of the GAN corresponds to a saddle point, at the end they obtained images with a resolution of up to 1024 x 1024 , useful for Full Field Digital Mammograms (know as FFDM).
The Figure 3.1 below, allows you to observe the difference between the original images and those completely generated by the GAN at the end of training. Despite the excellent visual results, there are still some artifacts in images, such as the bright or black spot and stretching effect. Therefore, the images are difficult for doctors to assess visually.

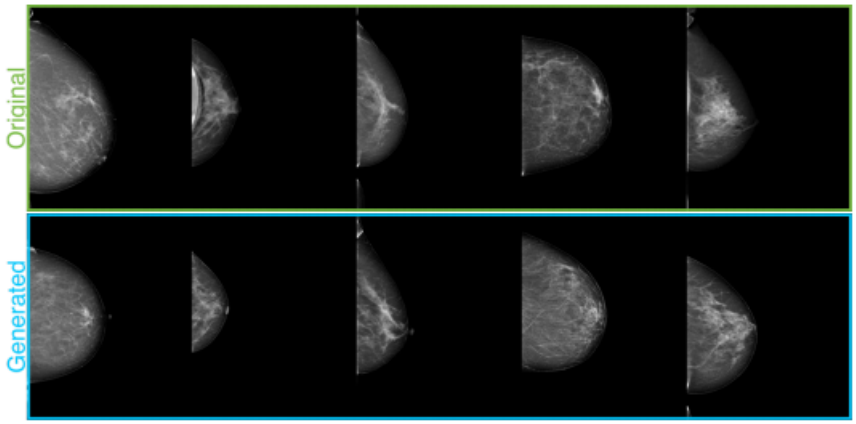


Figure 3.1: Randomly sampled examples of real and fake cranial-caudal views. Reprinted from [38]

3.2 Application of CycleGAN on medical images

The CyleGAN, essentially, consists of two generators to achieve the translation between two unpaired sets of images, as described in the original project [1]. There have been several studies that applied CyleGAN with medical images as data, for example in the field of image reconstruction and denoising, to increase image quality. Indeed, it was applied to images obtained during multi-phase coronary CT

angiography, to reduce the noise present when using a low dose without affecting the detailed texture and image quality. This was also possible by the implementation of the identity loss [39]. In [40] using cycle Wasserstein regression adversarial training, it was possible to obtain CT images with a super resolution, from images with a low resolution, maintaining anatomical information and reducing image noise. In this way, using generative models, it has been possible to overcome the problem related to the limited number of paired data available. However, to improve the results obtained, paired images were also used.
The use of CycleGAN has also become very popular for the synthesis of cross modality images, where from images obtained by magnetic resonance imaging (MRI) or CT, we can obtained images from other techniques. All this has numerous advantages: it avoids wasting extra time to carry out another screening, reducing the costs related to it. Moreover in this way it's possible to avoid the use of additional radiation to the patient. Further, because some techniques, like MRI permits the visualization mainly of soft tissues, unlike X-rays, which promote the visualization of bones and other tissues with a very high attenuation coefficient. Cross modality synthesis allows the visualization of different anatomical structures, which is more useful for diagnosis. However, it must be considered that this translation has a limited clinical importance. The patient often does not undergo both procedures. MRI is more specific for analysing small regions of the patient (with a higher resolution), whereas CT is more specific for large portions of the body (with a lower resolution).
Combining adversarial, cycle- consistency and voxel-wise losses, it is possible to obtain a generative model capable of converting 2D CT images of the brain into 2D images obtained through magnetic resonance imaging. To facilitate the diagnosis of eventual pathologies that are difficult to diagnose through CT [41]. This procedure can also be used for other applications such as CT-PET translation.
A difference process was also analysed, starting from images obtained with magnetic resonance imaging to obtain 2D images of CT [42]. Using two convolutional neural networks (CNN) as generators and two discriminators. Thanks to generative cycle consistency loss to obtain images as accurate as possible to the originals, with a dataset of 24 patients, who had submitted to both medical procedures to evaluate the new synthesized images. The results are visible in the Figure 3.2. In this study the images obtained with unpaired images were even better than using paired images.
While in [43], it was shown that simple image generation using cycle-consistency loss is not sufficient. In this paper the generator, composed of a CNN, is also trained with a shape consistency loss, in order to avoid geometric distortions in the formation of MRI 3D cardiovascular images. The generator is supported by a segmentor, which is fed by the images of both domains, and its training is alternated with that of the generator. To prevent the volumes of the cardiac images from be distorted. In this way it is possible a quantitative evaluation through segmentation and Dice

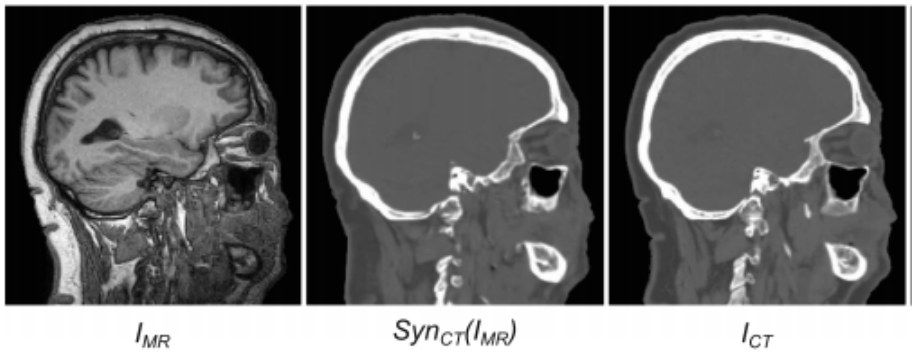


Figure 3.2: *From left to right* Input MR image, synthesized CT image and the real CT image. From [42]

.

loss, of the new volumes that are generated by the network (the volume-to-volume translation is much more difficult to implement than 2D translation).
However, the method proposed in the following chapters focuses on assessing the quality of the images obtained, without using other network models that have already been trained. In fact, a measure is proposed to objectively evaluate the distribution of the data produced by the generator.
In addition to this kind of translation, there are numerous other studies that use CycleGAN to create images that appear to be generated by positron emission tomography (PET) [44], to highlight particular organs or lesions. An additional translation that has been analysed is the possibility of translate from MRI images, obtained with T1, to images with T2. Although there is often the presence of both in medical imaging, only T2 may be more suitable for viewing different components like parenchyma, or T1 for better anatomical vision [45].
It's also possible to compare the performance of the CycleGAN with the UNIT model because of their similar construction and the possibility of working with unpaired medical images.
In [46], the two generative models are compared to a simple GAN (composed of a CNN), to transform MR brain images, with T1 and T2, into CT images and vice versa. Using quantitative comparisons, the simple generator is not able to produce very realistic images, while CycleGAN and UNIT, trained with adversarial loss also, are able to produce a lot of realistic images. The Mean Absolute Error (MAE) was applied to all three models, through comparison with ground truth. In particular, these two models demonstrate to be able to generate much more accurate medical images: from a quantitative point of view their results are extremely similar, with the CycleGAN showing a small improvement for T2 images, while the UNIT with T1.
On the other hand a model with a good score from a quantitative point of view does not necessarily generate realistic images for a radiologist. Images were then

subjected to a classifier to distinguish between T1 and T2. However, T2 shows more difficult to be classified because of its darker colour; the classifier is only one of many ways in which generative models can be qualitatively evaluated. Anyway, the model had trouble at the edges of the image and in the portions of the brain where there was a greater variation in intensity. The greatest difficult, in the paper, is the use of parameters or scores to determine the quality of the data obtained from the generator.
As demonstrated in [47], using just one GAN to work on complex images, like retinal fundi images, is not enough for stability and noise problems. For this reason, two GANs models were used, to obtain in the first phase the segmentation masks, while in the second phase create the corresponding retinal fundi images. While the discriminator is a common CNN, with two convolutional layers, the generator is a deeper neural network, with more layers, fully convolutional and without pooling layers. This configuration has demonstrated to be more stable and efficient. To evaluate the reliability of the generated data, they were used to train a U-Net segmentor, to create a segmentation mask from retinal fundi.

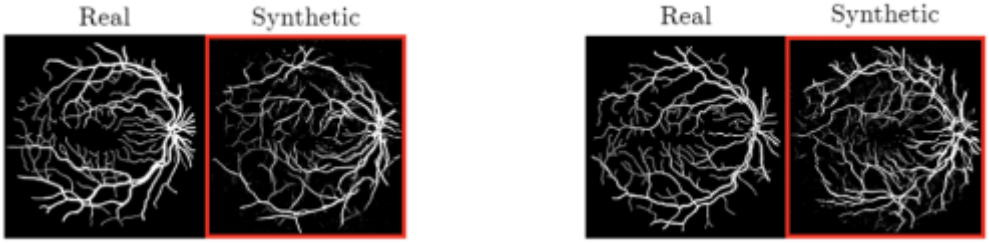


Figure 3.3: Comparison of real and fake segmentation mask of vessel tree in the retina. Adapted from [47]

Also in this case there were difficulties to obtain quantitative results directly on the synthesised images generated by the two GANs. It was decided to directly compare the U-Net's training results with another database, using the Kullback-Leibler divergence score as a quality factor. This paper highlights, not only how the use of two GANs allows more realistic results, but also underlines the difficulty and the different methods proposed to analyse directly, the images generated.
In a recent study the use of CycleGAN to transform contrast CT abdomen images into non-contrast CT was evaluated, these images are then used for data augmentation during the training of a segmentation network [48]. Therefore, the generative model was build in order to create realistic images as much as possible, to train a network with a smaller number of images available (linked to the common problem of the lack of large datasets of medical images).
It has been shown that the U-Net segmentor, thanks to the increase of data during training phase, performed better. The U-Net is trained with a small batch size of 16.

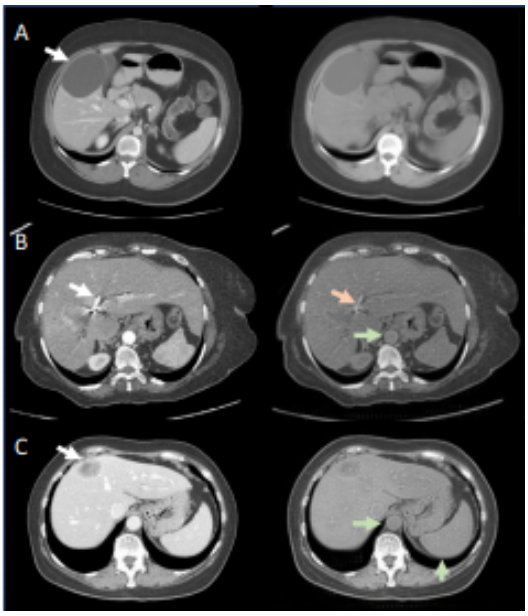


Figure 3.4: *From left to right.* Examples of true contrast CT images and synthetic CT images without contrast. Image taken from [48]

The network shows an increase in the quality of kidney, liver and spleen segmentation, estimated by the Dice loss; which increased especially when it was trained with the images generated by the model while the volume estimation error decreases from 0.450, for the classic augmentation method, to 0.189 for the use of CycleGAN. It was demonstrated that the augmentation with generated data, increase more the performance in comparison to the standard augmentation method and histogram equalization. Even simple augmentation using spatial transformation is not the optimal solution with CT images, where attenuation or modification depends on various biological factors. This result is also linked to the strong local differences related to intravenous contrast injection, where the intensity variation changes in different organs (especially in the kidneys where it accumulates) and in different patients. The model is limited only with 2D slices and not 3D volumes.
However, CycleGAN has showed some difficulties in the preservation of boundaries structures in images where there is a large close presence of muscles, bones and tendons, such as the pelvic region [49].
In [50], have attempted to translate from MRI images to the corresponding CT images in this region, to emphasize especially the bone component useful for diagnostic examinations. They added a loss function, called gradient consistency loss, which evaluates the consistency of each pixel of the image gradient, by evaluating the difference between the actual image and the image generated from the network. The improvements of performance are highlighted by the improved DICE score, with a U-Net, to segment anatomical components like the femur (where the new

images have kept their shape near the femoral head and adductor muscle), the gluteus medius and the pelvis.
A similar approach has been applied in the pelvic region by Kida et al. [51], where it was pointed out that an incorrect initialisation of weights on the neural network, induced the generator to produce completely distorted images and to learn an incorrect function mapping. In this case the translation was from a cone-beam CT (where X-rays have a cone shape) to a planning CT.
One of the most recent studies is the one presented this year at MICCAI, which is a conference organized for scientists in the areas of Medical Image Computing and Computer Assisted Interventions. They try to add contrast in CT abdomen images, starting from images without contrast [52]. Underlining the importance of preserving small elements during the translation between domains, like little calcified plaques present in the renal aorta and pelvic arteries.
This study was characterized by the introduction of a network, based on a shared latent variables from a Gaussian mixture model. In this way the model is applied to a CycleGAN and to a UNIT net, where two models of variational auto-encoders, which compose the two generators, and a common latent space z is present. The Figure 3.5 describe the model.

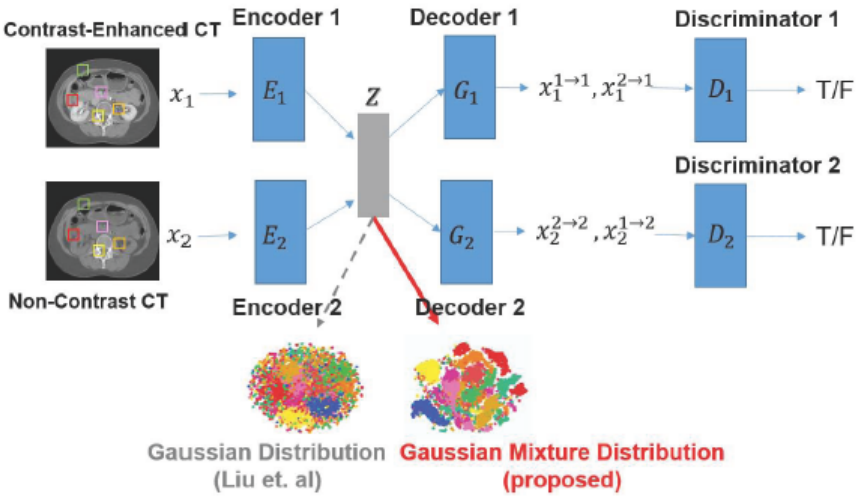


Figure 3.5: The model presents a common latent space, where variables lie in a Gaussian distribution. Reprinted from [52]

From each of the 512 x 512 medical images, different small patches (32 x 32) are extracted, which present the many features extracted from the original image. From each images of both domains, the different variables are divided into K clusters inside z, each with a Gaussian distribution with an average and standard deviation. The optimal number of clusters identified for that specific dataset is $K = 25$.

The images are then reconstructed with the help of two generators, two variational auto-encoders (VAEs). In particular the VAE loss is based on the KL divergence, if the distribution in z diversifies too much after the priority distribution. To improve the quality of the images, a cycle consistency constraint is inserted, where the net try to resemble the images in their original domain.
Also in this case, the quantitative performance evaluation is applied on another trained network to identify and segment calcified plaques in renal vessel and pancreas. The images used for training are 140 unpaired CT images. For both situations the trained net has a better Dice score for segmentation and a greater focus on the preservation of fine structures. This is possible thanks to the creation of clusters in the common latent space and the presence of a cycle loss. This allowed the network to identify groups of images with similar characteristics.
For the model that has been developed in the last few months, a similar method has been applied. It is based on the identification and distinction of different sections of the patients' bodies, and on the focus of the renal component. This method exploits a common feature of medical image databases, which is that they consist of unpaired images. CycleGAN, in fact, is developed for this type of data. In addition, in the final part of the study, an attempt is made to develop a method for evaluating the data generated. That it can also be applied to other studies. Indeed, in contrast to the projects described before, which were based on the application of other neural network models; we use metric and visual analysis, in order to make it easier to objectively compare the generated data.
Another important aspect that was taken into consideration was the use of only healthy patients to realise the training and test set, to homogenise the input data. As described in [53], the transformation of images from one domain to another can produce distortions or hallucinations in the distribution of output data. This happens when patients with different health conditions, leading the generator to produce wrong elements in the generated image. For this reason, the output images are not subject to direct interpretation by the doctor, as this could lead to wrong diagnoses. A more adequate use is, therefore, the training of deep networks.

Chapter 4

The organization of CT images

The entire project describes a method to transform normal abdominal CT images into contrast-enhanced CT images, and vice versa, using a Cycle-Consistent Adversarial Network (CycleGAN). In the first part, the chapter describes the databases, from where the project data are extracted: highlighting the characteristics and the number of patients selected.
The second, and final, section describes the pre-processing of the data in order to improve the quality of the images and facilitate the identification of key features by the Neural Networks.

4.1 Division of data

The images were obtained from the Cancer Imaging Archive (TCIA), which contains a large archive of medical images of different diseases available for free download to the public.
The aim is to obtain several generic image databases that are suitable for the study, which are acquired directly from the CT scanner and were not previously processed or modified.
After a careful selection, three databases of abdomen CT images were chosen, whose number of patients involved is described in the Table 4.1.
The first database presents scans of 825 patients, who were analysed for the possible presence of polyps, this database is characterised by the presence of a dilated colon. The CT scans were all obtained without the insertion of contrast inside the patient. It was divided according to the presence (or absence) of polyps and their dimension. It was decided to work only with scans of healthy patients, because the presence of tumour cells or other abdominal pathologies would have produced greater variability in the images, making it more difficult for GANs to generate new images. This was done for all the databases. For this reason of the 825 patients present, only 213 healthy elements were kept.

The second database consists of 89 abdomen CT scans, where the position of the lymph nodes has been previously marked, but the patients don't present any abdominal pathology. The images of the relative scans are characterised by the presence of contrast inserted with arterial phase.
The third, and the last, database presents 82 CT scans of contrast images, acquired 70 seconds after intravenous contrast injection in portal-venous. The database was created using the pancreas as a reference for possible donors. In this case the patients do not present also major abdominal pathologies or lesions. Two different domains were obtained from the three starting databases: domain A, composed of the last two sets, is made up only of images with the contrast injected; while domain B, is made up of normal CT scans. More specifically, only contrast inserted in early arterial injection phase, where the kidneys and major renal vessels are more visible, was analyzed. All images have an high resolution of 512 x 512 and are in DICOM format.

Domain	Database	Female	Male	Unknown
B	CT Colonography	103	93	17
A	CT LymphNodes	43	46	0
A	CT Pancreas	0	0	82

Table 4.1: Number of patient for each database

The images were all converted from DICOM to NIfTi format. The first one, in fact, despite it is very flexible, powerful and have a great interoperability between different software, is not very efficient for image and signal processing [54]. In addition, a single 3D volume, stored as a series of 2D slices, can occupy a considerable amount of space.
On the other hand, the NIfTi format, it is simpler than DICOM, stores a volume in a single file, allowing to occupy less space. All the related metadata contained in the original DICOM files were transcribed into appropriate JSON files (called headers), for each patient's 3D volume.
The conversion was possible thanks to MRIcroGL, which allows to view 2D slices and renderings of your brain imaging data.
All patients were then divided, to facilitate the selection, according to the hierarchical scheme presented below (Figure 4.1). Through the analysis of the metadata, only patients in the supine position were preserved, while those in the prone position were removed. That's because the first kind of position represents the standard CT scan protocol, for this reason is more common. Similarly, patients whose scans were acquired in lateral decubitus, left and right, view (indicates with HFDL and HFDR) were also removed.
This selection, applied mainly for the first database, was performed to try to keep

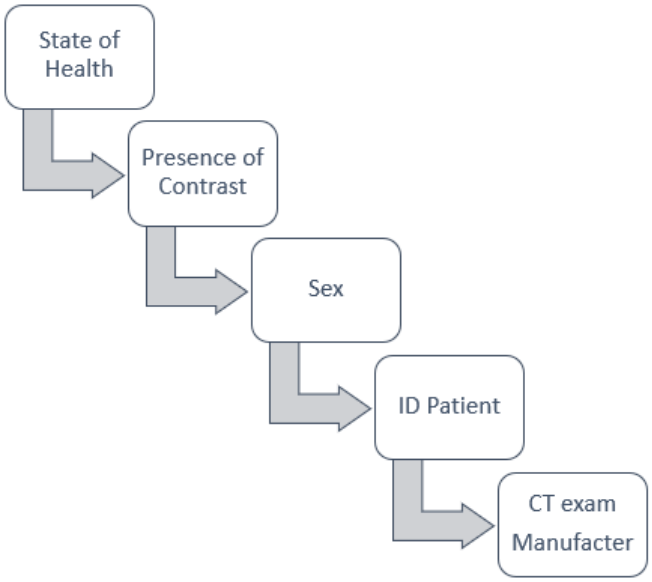


Figure 4.1: Scheme of the organisation of patients in the dataset

the disposition of the different anatomical components as homogeneous and invariant as possible. Subsequently, the position in which the patient's feet are facing the front of the imaging equipment (i.e. feet entering the front of the equipment) was taken as standard (indicates with FFS). In order to avoid losing too many patients, all those who were positioned toward the front of the imaging equipment with the head (Head First Supine position) were converted into arrays and then inverted, along the third dimension, to align with the others. The Figure 4.2 shows three samples taken from the each databases.

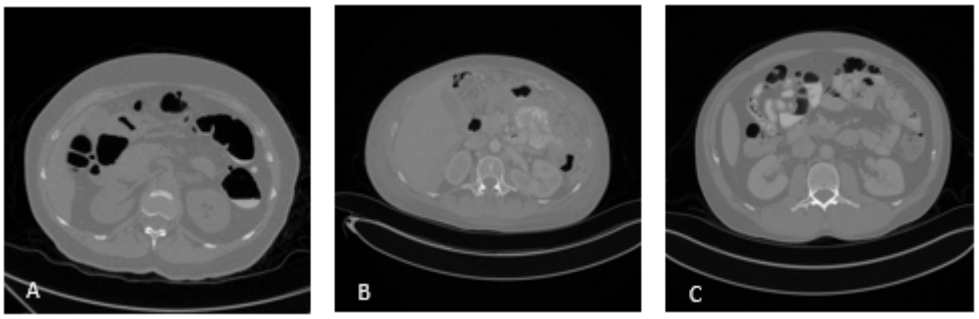


Figure 4.2: (A) Non-contrast image, from CT Colonography. (B) Contrast-enhanced (with arterial phase) image, from CT LymphNodes. (C) Contrast-enhanced (with arterial phase) image, from CT Pancreas

4.2 Pre-processing of data

The purpose of pre-processing is to improve the data furnished, to facilitate identification of key features and the learning of the network.
First, the pixel size distribution, for both domains, was calculated by analysing each NIfTi file. The Figure 4.3 below represents the distribution of the data. The third dimension indicates the 3D space between two consecutive slices of the same patient, which is based on the frequency with which they were obtained. The images are then resampled, often the median value for each axis of the spacing is taken as the standard value. However, this choice may cause loss of information or undesirable distortions. The resampling of all images is performed by a third-order

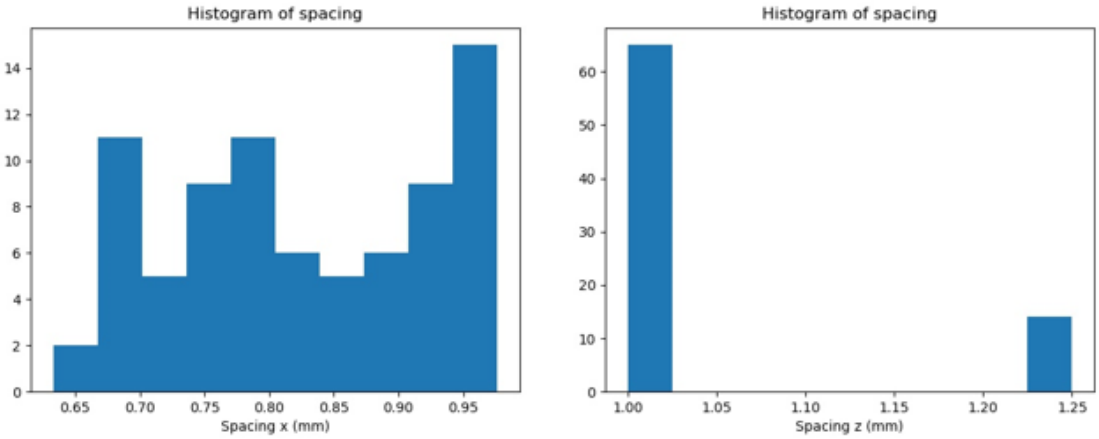


Figure 4.3: Distribution of pixel sizes of X and Z axes of domain A.

spline interpolation, whose zoom factor is calculated as the ratio between the pixel size of the corresponding image and the total average [55]. The resampling is applied for each axis, in order to prepare for a future application with 3D volumes also.

Domain	X	Y	Z
A-With Contrast	0.8	0.8	1.0
B-No Contrast	0.7	0.7	0.8

Table 4.2: Average pixel size (mm) for each axis

This interpolating function for medical images, has been proven to produce: fewer errors, a minor loss of information and does not apply a large smoothed effect, compared to other interpolating functions such as the nearest neighbour or linear [56].

Subsequently, to facilitate the visualisation of interesting anatomical components, all images, with and without contrast, are clipped (this is possible thanks to the conversion of the data into numpy arrays). In the CT images, the intensity of the different pixels depends on the chemical and physical properties of the different tissues.
All values for both domains are then kept within the range: [-70, 303].
This operation makes it easier to identify the bone component, which stands out more in the image, and also the contrast present in the vessels and kidney components.
The next step is to create a function to remove the presence of the "white bar", visible in the first column of Figure 4.4. This artifact is caused by the presence of the machine itself. This function keeps the pixels with an intensity different from the background which are close to each other, i.e. only those of the abdominal component (Figure 4.4). In this way the generator is not disturbed by unnecessary element when it is identifying the fundamental features.

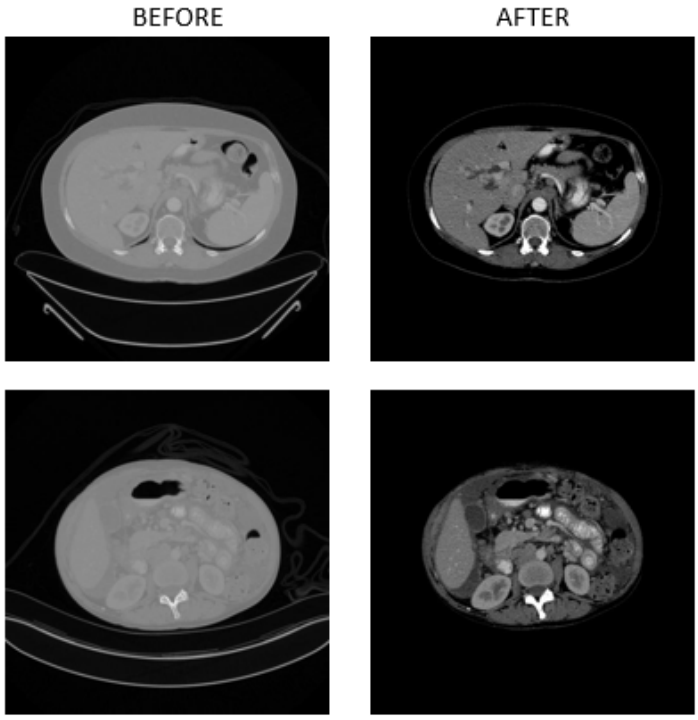


Figure 4.4: The removing of the artifacts in the background.

The data, subsequently, are normalised with z-scoring, where each image is modified by subtracting the mean and dividing by the standard deviation of the dataset. The values identified are: a mean of -42 and a standard deviation of 63. For the first two databases, used to compose the two domains, the same values are applied

for the normalisation [55]. Usually GANs take as input data, normalised values between -1 and 1, this was done by taking the maximum value of for each 2D image. Because this normalization did not lead to optimal results, it was also tried not to apply it. Finally, at the end of this process, choosing to work in 2D, all images are resized from 512 x 512 to 128 x 128. These are the input for the generative model. The reduction of size provokes an inevitable loss of information in the image, but this was applied to make the training computationally less expensive and faster.

Chapter 5

Proposed method: CT image translation using CycleGAN

This chapter describes the application of the generative model to obtain contrast-enhanced CT from contrast-free images. The translation is also in the opposite direction.
The chapter describes the principal characteristics of the model developed and the main problems observed during the project. In particular, the last sections describe the different strategies applied to improve the data produced by the generator and to increase the stability of the GANs.

5.1 Parameters of CycleGAN

First of all, an apposite dataset was written, with Pytorch, able to provide the two generators with 2D random images, relative to the two different domains, which are converted in vectors and provided as input.
As mentioned in the previous chapter, the complete loss of the elaborated model is described in the 5.1.

$$\begin{aligned}\mathcal{L}\left(G, D_{\mathrm{X}}, D_{\mathrm{Y}}, F\right) = \mathcal{L}_{GAN}\left(G, D_{\mathrm{Y}}, X, Y\right) + \mathcal{L}_{GAN}\left(F, D_{\mathrm{X}}, Y, X\right) + \\ \lambda_1(\mathcal{L}_{cyc}\left(G, F\right) + \mathcal{L}_{cyc}\left(F, G\right)) + \lambda_2(\mathcal{L}_{idt}\left(F, X\right) + \mathcal{L}_{idt}\left(G, Y\right))\end{aligned} \tag{5.1}$$

In the 5.1 the two generators are indicated by G and F, and the two set of images are described by X and Y. The λ_1 is the weight for the cycle loss, and λ_2 is the weight for the identity loss, which is preferable to impose with a value that is less than 1/10 of the cycle loss, since this affects the intensity of the pixels and must not excessively modify the morphology of the generated image. Several tests were carried out to identify the best combination of these two weights, in the end we chose as λ_2 equal to 0.5 and λ_1 equal to 10. For identity and cycle loss, it must be considered that they involve the two generators present to promote training in both

directions. For this reason their contribution is given by the sum of both directions. The performance of the generator is evaluated with the mean squared error. This criterion is calculated between the output of the discriminator and a matrix composed of ones, corresponding to the labels of a real image.
In this way the generator modifies the gradients of the network to reduce this difference, which indicates that the discriminator is being fooled and interprets the image created by the generator as real. While the Identity and Cycle loss are evaluated with the mean absolute error (MAE).

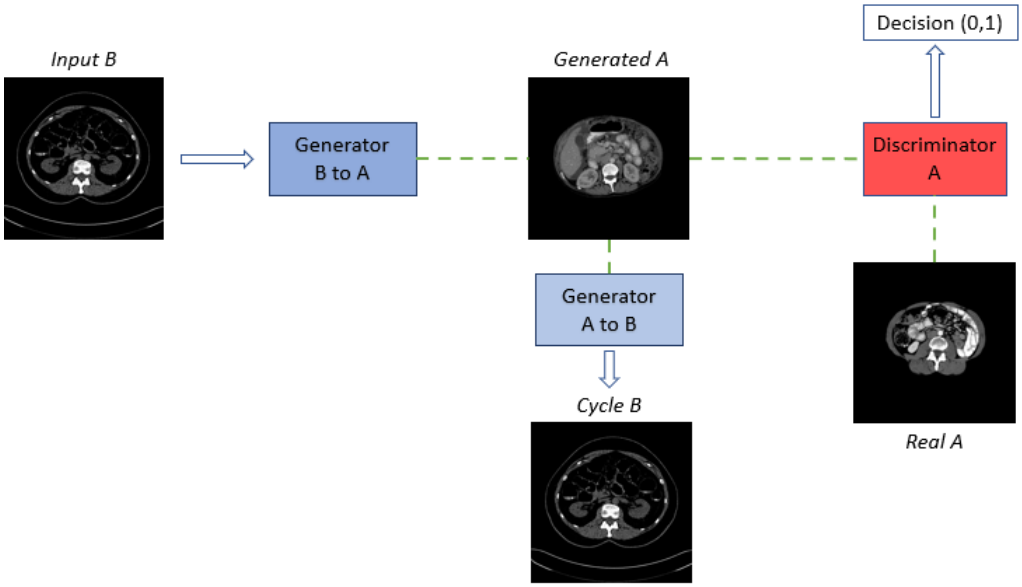


Figure 5.1: Pipeline of the main elements of developed model

The Figure 5.1 shows the organisation of the main elements of the generative model: starting from an image taken from domain B, through the generator we obtain a corresponding image with the inserted contrast. This last one is used as input for the discriminator besides a real image taken directly from the database. At the same time the obtained image is used as input to realise the Cycle-Consistent loss. This organisation is repeated symmetrically in the opposite direction, so that both directions of translation are trained.
Among the various other hyper-parameters initially set, the training phase consists of 200 epochs, while the learning rate trend (Figure 5.2) starts at $2 * 10^{-5}$ and then reduces linearly to 0, from the 100th epoch. Based on the CycleGAN model described in [1], the learning rate was initially set to $2 * 10^{-4}$. This value is considered optimal for different types of data, but it was observed that for this type of medical images it was too high. For this reason it produced undesirable divergent behaviour in the loss function and an unstable training process. Exponential reductions were also not particularly effective, as they led to a too fast reduction in the learning

rate. In the same way, in order to favour the generation of good quality images,

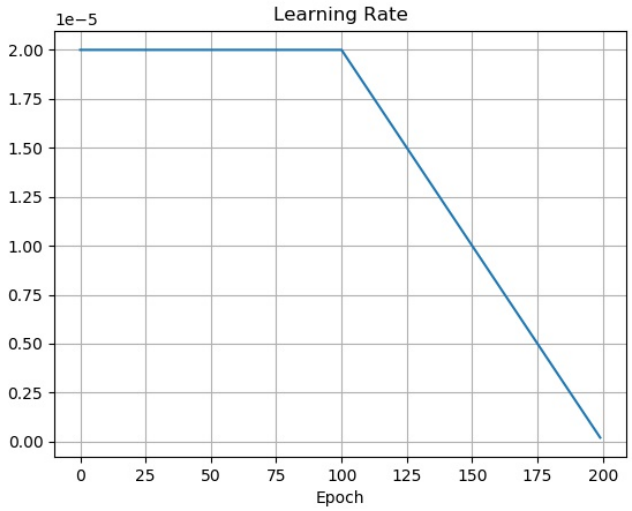


Figure 5.2: Learning rate trend

the size of the images was reduced to 128 x 128, and a first dataset was created consisting only of female patients, to avoid additional variation linked to different anatomical elements present.
It was observed that different patients, present as input to the generator a high variation related to the different anatomical components that were displayed. Indeed, these anatomical components characterise the images with very different features: such as a high bone component is visible in the lower part of the pelvis or the lungs, which, for the low attenuation coefficient of air, are characterised by a much lower pixel intensity.
To solve this limitation it was decided to select a reduced number of slices. For domain A, after manually analysing 10 patients, it was decided to take only about 150 images for each. Identifying the mid-point where the lungs and anterior iliac spines begin to appear.
On the other hand, for the domain B, a range of 175 image is taken for the same section, the difference is related to a smaller distance between two adjacent acquired images (visible in the Table 4.2). The number of data available is:

- For domain A, consisting of the database CT LymphNodes, 79 patients (43 women and 36 men) are considered. The number of data used for the training phase is 11850.

- For domain B, consisting of the database CT Colonography, 68 women are considered. The number of data used for the training phase is 11900.

Initially, the input of the model was composed of a single image, then it was decided to use input with a batch size of 10. The use of a high batch size and of a batch normalization, facilitates the stabilization during the training phase for the discriminator and the generator. However it is not applied on the last layer of the generator and on the first layer of the discriminator [57]. The architecture of the two Deep NNs is described in the section 2.8.4. With these characteristics, the U-Net and ResNET were tested as generators. The Figure 5.3 compares the losses of the generators, built with two different models. It can be seen that both have a high instability initially, with the U-Net showing several peaks. Finally, both models converge towards the same value. However, the values obtained are very high. The Identity Loss presents a highly unstable trend, though the values reached by it are extremely lower than those of the generator and the Cycle Loss (Figure 5.4). The latter shows an extremely unstable trend. The U-net, in particular, exhibits a much greater range of variation than ResNET.
However, the performance of the different losses during the training phase is not the only way of identifying the best combination of hyper parameters and assessing the model.This represents one of the main limitations of generative models based on GANs. In addition to a quantitative analysis, a visual examination of the generated image quality was applied. The U-Net, despite having a similar pattern to the ResNET, produced a very different distribution of data. Generating highly unrealistic images. The Figure 5.5 shows examples obtained from the generative model, trying to transform images from domain B, to domain A. In this respect it is noted that there is a marked increase in pixel intensity in the whole image. This change also affects the background. There is a greater visibility of the kidney component, but the other anatomical components also undergo a big variation. Another problem is the alteration of the anatomical component, which can be very dangerous for diagnostic purposes. In particular, in the U-Net, it is noted that the generator particularly modifies the liver, adding non-existent components.
In order to improve the results obtained so far, different modifications have been added to obtain better results at the end.

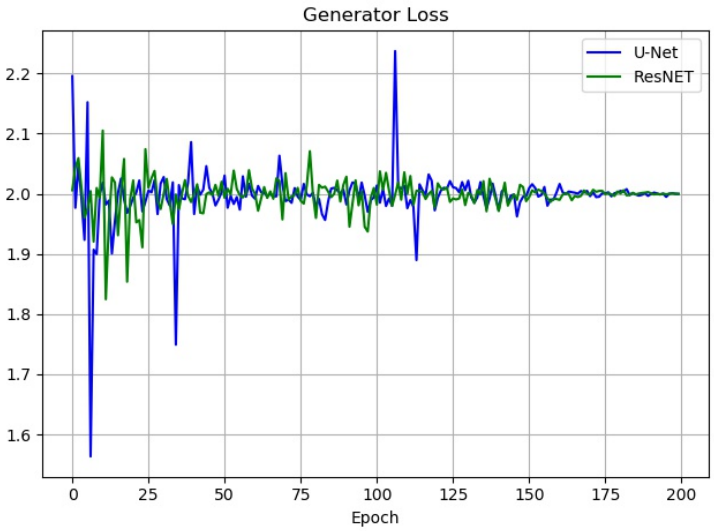


Figure 5.3: The trend of the Generator Loss.

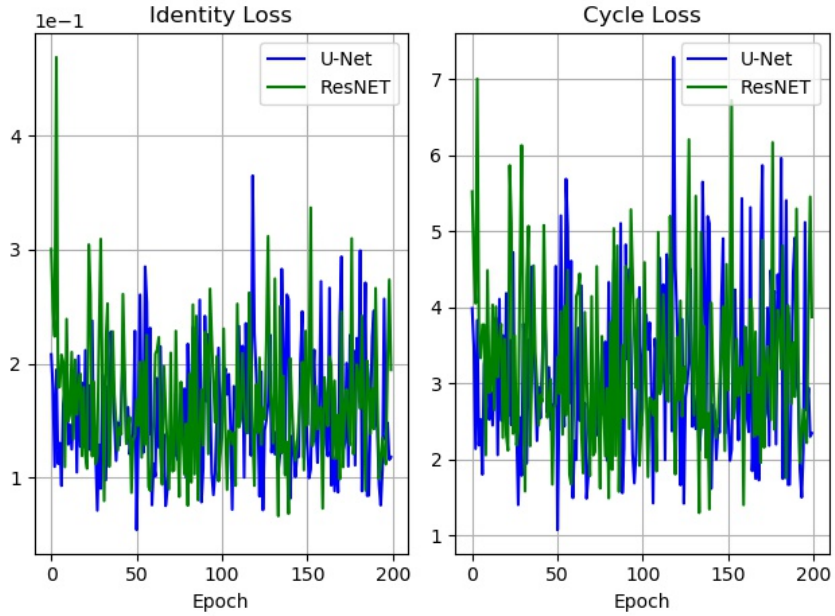


Figure 5.4: *From left to right:* Identity and Cycle Loss trends.

5.2 The implementation of Wasserstein loss

GAN is based on the discriminator that tries to distinguish real images and images created by the generator.
The discriminator used in the model divides input images into different patches and assigns as a value 0, if it identifies it as a real image, otherwise it assigns 1 if

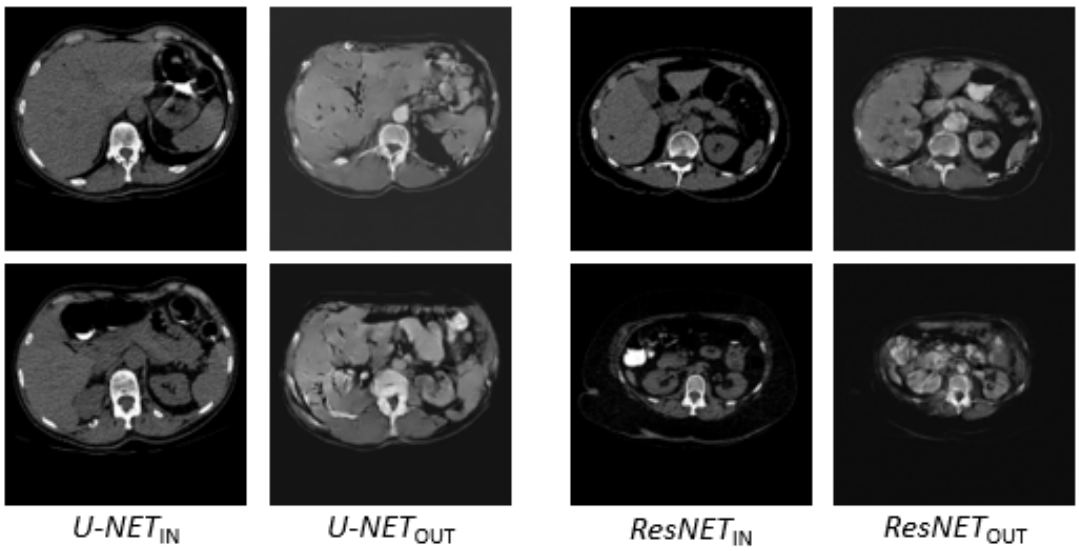


Figure 5.5: *From left to right:* Examples of true simple CT images and synthetic CT images with contrast, generated by the U-Net and by the ResNET.

it believes it was created by the generator. In the same way, the image provided as input to the discriminator, determines: a matrix with only 0, if it has been taken from the original dataset, or with a matrix filled with ones, in the case where it is taken directly from the output of the generator. The difference between this latter matrix and the one based on the discriminator's "deductions" determines the discrimination's loss. In this way, the discriminator tries to classify the probability of the furnished image as real or false, which encourages the generator to produce more realistic data to fool the discriminator.
This difference is initially calculated with the squared L2 norm (a criterion that measures the mean squared error). At the beginning, his criterion was considered more reliable, although in [1] the CycleGAN was constructed using the Binary Cross Entropy,calculated between the target and the output. The latter loss, however, led to the generation of images of low quality. They were analyzed by a visual inspection.
Through a careful analysis of the most recent papers with generative models, it was found that the implementation of the Wasserstein loss represents one of the best solutions for the generation of good quality images. Furthermore, it favours the stability of the network in the training phase (which is compromised by the training of both neural networks present) and convergence.
The discriminator does not assigns a probability, but it scores the realness or fakeness of input data. The two generators are trained to minimise the distance between the distribution of the real data taken from the dataset and the distribution obtained from the generated images. This distance is defined Wasserstein distance:

which is continuous, differentiable and gives a linear gradient, even when the discriminator is well trained. It is described as the shortest average distance necessary to move the probability, from one distribution to another. Indeed, it has been noted in the previous cases, the discriminator learned very quickly to distinguish real from fake images, and fails to provide useful gradient information to update the corresponding generator [27].
However, due to the Wasserstein loss, the discriminator will converge to a linear function, but continuing to provide useful gradients for the model. Moreover, one of the main problems with the training of GANs is that the loss trend does not always correspond to the quality of the images generated. These must be checked, at the end of each process, by a visual analysis. Through the implementation of the Wasserstein GAN, it is possible to promote the correlation between the loss of the discriminator and the quality of the data produced.
To implement it, a linear activation function was added to the last layer of the discriminator. In [27] to train the discriminator the weights are clamped within a specific range [-c, c] after each update, in order to have a compact space. In addition the discriminator is updated more times than the generator in each iteration. This solution, however, was not suitable for the developed network, because the discriminator obtains immediately much better results than the generator.
The restriction of the weights can cause the vanishing gradients and undesired behaviour. This happens when a too small range [-c, c] is chosen. It also makes the model extremely sensitive to hyper parameter c.
A small variant based on the gradient penalty was implemented, as proposed in [58]. In this case, instead of clipping the weights of the discriminator, a limitation is imposed on their length. The loss of the discriminator is described in 5.2.

$$L_D = D(\mathbf{x}) - D(G(\mathbf{z})) + \gamma(\|\nabla_{\hat{\mathbf{x}}} D(\hat{\mathbf{x}})\| - 1)^2 \tag{5.2}$$

The first two elements of the equation 5.2 can be found in [27] and [58]. They indicate that the discriminator wants to separate as much as possible the real data and the data created by the generator, increasing the difference between them. The output values will be as large as possible for fake samples and as small as possible for samples from the original dataset. In the formula, X is the input of discriminator and Z is the input of generator. The third element is the gradient penalty. It takes the gradient at $\hat{x}$, a randomly weighted average between a fake and a real sample from the dataset (5.3).

$$\hat{\mathbf{x}} = \epsilon\mathbf{x} + (1 - \epsilon)G(\mathbf{z}) \tag{5.3}$$

Data are interpolated,and ϵ is a random value, chosen between 0 and 1. These points obtained, should have a gradient norm of 1. So instead of applying clipping, the gradient penalty punishes the network if the gradient norm deviates from its desired norm value 1. For the built model λ in 5.2 is set to 10. While for the loss

of the generator the equation 5.4 is applied. The generator tries to minimize it, to fool the discriminator.

$$L_G = D(G(\mathbf{z})) \tag{5.4}$$

In addition, the activation function in the last layer of the NNs, of both generators, has been modified. Initially, the hyperbolic tangent function (Tanh) was simply applied, with an output range of [-1, 1]. However, multiplying this function by 6, the range was increased to [-6, 6]. In fact, the values inside the image, due to the Z score and the clipping applied in the pre-processing phase, make the pixels' value limited within the range [-0.5, 5.5]. In this way, by increasing the range of the activation function, the pixels can be modified by the generator, allowing it to reach any value within the range.
Another important change, that has led to a considerable improvement in results, is the use of the input image as a template on which the generator can generate new data. It is added directly to the output. In this way the two generators do not have to build the image completely from zero, but they focus more on the main changes they want to apply to their output. This template is applied in both directions.

5.3 Application of binary masks

Thanks to a a visual evaluation of the images saved at random epochs during the training phase, it was noted that the generator did not only modify the abdominal components visible in the CT image, but also corrected the intensity of the pixels in the background. This was due to possible noise in the background, which influenced the network.
A binary mask was implemented, in order to focus the neural network only on the anatomical component. This takes only the abdominal component, making in this way that the modifications related to the intensity of the pixels concern only this section. This solution is visible in different papers, where Generative Adversarial Networks are implemented. In [59] for example a binary mask is generated, in this way only the surgical mask of the subject is taken, leaving out the rest of the face. The mask is realized thanks to the implementation of a function, which identifies the pixels with an intensity greater than the minimum value of the image (corresponding to the background). The edges of the pixel regions are then gradually expanded due to the effect of dilation. Their area is extended, while the blanks in these regions become smaller. Thus a binary mask is created whose foreground pixel corresponds only to the anatomical component (Figure 5.6).
The mask is calculated for each input image. It is then multiplied in the last layer, where the Tanh activation function is applied. Thereby all the variations present in the background are set to zero and only pixels relating to the anatomical component are kept intact. Consequently, the net focuses only on the abdominal component, modifying it.

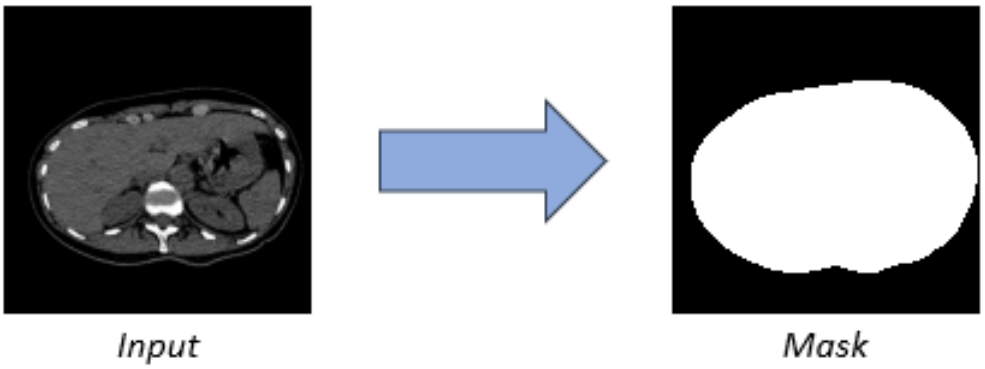


Figure 5.6: Randomly sampled example of a mask obtained from an input image.

5.4 Identification of the renal section

The main purpose of the study is to train the generator to artificially insert the effect related to the injection of contrast agents in the patients. It is based on the objective of generating a data distribution that corresponds as much as possible to a dataset composed of abdominal CT images of patients in whom contrast agent has been inserted to favour the visualisation of the renal component.
Previously, it were described the main improvements that were applied to the NNs that constitute the CycleGAN model. We tried to homogenize as much as possible the type of images used as input of the two generators: to favour the visualization of the effect related to the contrast medium and, then, a binary mask was applied to focus only on the abdominal component.
However, it was noted that the images provided as input for the two generators had very different characteristics. Initially, the number of slices for each patient was limited, thanks to the manual identification of a specific range. Which starts from the end of the lungs until the formation of the hip. However, the average position was calculated on a limited number of patients corresponding to 10% of all those available. For this reason, different patients had different acquisition ranges and different characteristics. These were partly due to the presence of different machines. These differences are visible in both domains.
However, a new approach was needed to identify only those anatomical components for each patient in the study. The difference between the images, due to the insertion of contrast, is only visible at the level of the kidneys and the most important renal vessels. Nevertheless, using as input images of different anatomical components, such as the pancreas or stomach, confuses the model and makes it tend to add the contrast effect to these components also.

On the other hand, because the images of the two domains at the level of the stomach (or liver) are very similar, the ResNET has difficulty to identify different features and applying them the contrast effect. Therefore, it is necessary to identify and select different sections for each patient, based on the main anatomical components. A similar approach can be seen in [52], where to facilitate the formation of new abdominal images by exploiting CyleGAN, the different images are divided into clusters according to the features present. To avoid excessively modifying the model, a different technique was adopted.
A function was developed to calculate, for each patient slice, the number of pixels with a minimum value. In a grey-scale image the minimum value corresponds to black. From the graphs obtained for each patient we observe a decisive increase in the number of black pixels, in the last slices. This increase is due to the presence of the lungs, which are filled with air. This area, having a very low X-ray attenuation coefficient, does not absorb the incident energy and form black spaces in the image.

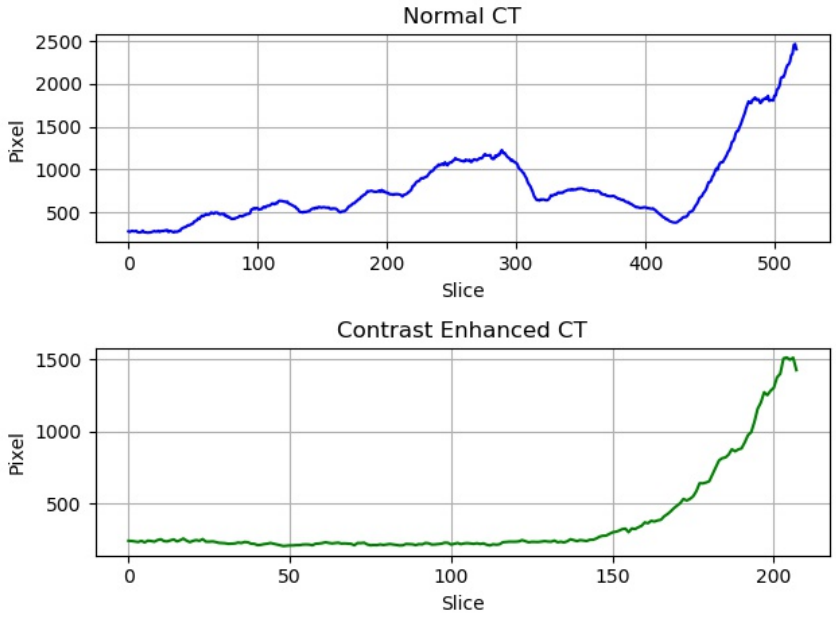


Figure 5.7: Evolution of the number of black pixels in two samples: from domain B (*upper*) and A (*lower*).

As can be seen in 5.7, patients belonging to different domains present similar graphs. It is necessary point out that hat domain A, from now on, is composed only of patients taken from the PANCREAS CT database. This is indispensable because it has been observed that both images taken from the latter database and from CT Colonography, are marked by a full colon. Probably due to the lack of intestinal cleaning prior to the clinical examination. In this way, the homogeneity between

the two domains increases.
For each patient the corresponding graph is obtained, where it is visible an increase in the number of black pixels in the last 100 slices, due to the presence of the lungs. In this section a landmark is placed at the maximum point which indicates the average height of the lungs, before the cardiac component appears.
Therefore from each landmark four specific sections are determined, which correspond to the presence of certain anatomical components. This division and the length of each section is described in the Table 5.1.

Database	Section 1	Section 2	Section 3	Section 4
CT Pancreas	40	60	60	120
CT Colonography	50	70	70	150

Table 5.1: Number of slices of the four sections identified for each database

The first section starts at the landmark and includes the lungs. The second section shows the liver and part of the kidneys. The third section includes the colon and the kidneys, until they disappear. The fourth section shows the bladder. The difference in numbers between the two domains, is determined by the thickness of each slice acquired. Furthermore, the fourth section is not always well defined, as some acquisitions on different patients end before it.
It is necessary to point out that despite this more precise identification and viewing of different anatomical sections, there is still a discrete variability between patients. This is also present because we did not want to excessively select and process the data provided. Because the purpose is to develop a model able of working with highly variable data directly provided by the CT scanner. Thus we try to make the identification of fundamental features of the data distributions more stable, more specific for the analysed body section. here the effects of the contrast agent are more visible
This way produces a specific division for each patient used in training.

- For domain A, consisting of 77 patients from the Pancreas CT database, 9240 input images are used, which are taken only from sections 2 and 3.
- For domain B, consisting of 69 patients from CT Colonography, 9660 input images are used, from the sections 2 and 3.

Compared to previous versions, there is a decrease in the number of data, at the same time there is a greater precision about the anatomical section of interest. These techniques applied to the generative model allowed for greater stability and improvement of the final results, which are described and evaluated in the next chapter.

5.5 Application of 2.5D Neural Networks

One of the main problems noted in CycleGAN is the difficulty that the model has in identifying the exact portion of the image in which the contrast is inserted, modifying the values. Indeed, it was observed that the increase in pixel intensity, during the translation from the B domain to A, is concentrated in the renal section, but, at the same time, involves other abdominal components also.
One way forward is to move from the simple 2D technique to 2.5D. This consists in using as input, not a 2D grey-scale image (distinguished by a single channel), but three consecutive slices: each of which is inserted into one of the three specific channels, like a classic RGB image. Accordingly, the number of input and output channels of the generators, and the input of the discriminator are modified. This technique is known as 2.5D or pseudo 3D.
In this way we try to provide a minimum of three-dimensional information to the generative model. It, along these lines, can more easily identify the portion of the body subject to the change due to the contrast agent.
However, this technique has led to a decrease in the quality of the images produced. This is attributed to the use of axial images. More precisely, it is due to the anatomical differences between two consecutive slices and the space between them. These differences are related to the acquisition time of the CT scan. This makes it more difficult for the generator to correctly identify the anatomical portions of interest. Different results could be obtained with the use of coronal images. Where the space between 2 consecutive slices is smaller and, for this reason, the anatomical difference present between them is reduced. In fact, the Table 4.2 shows that the third dimension is higher in the selected databases. This characteristic leads to a worse quality of images created by the two generators.
The Figure 5.8 shows the results of this technique obtained during the test phase. As can be seen, the generator (composed of the U-Net), in the attempt to add contrast, slightly increases the intensity of the pixels, not in a significant and precise way. The relative residual map shows that the greatest increase is in the spine, while the renal component undergoes a slight alteration. The residual map is created by subtracting the output with the input data, in order to identify and quantify the anatomical components that undergo major changes in the image.
In the case of ResNET, however, the generator is unable to apply any substantial change in the output image, except only in the rachis. With the application of 2.5D, the precision of the generative model does not improve, but actually deteriorates.
In the next chapter, the results are described, analysed and commented, from a qualitatively and quantitatively point of view.

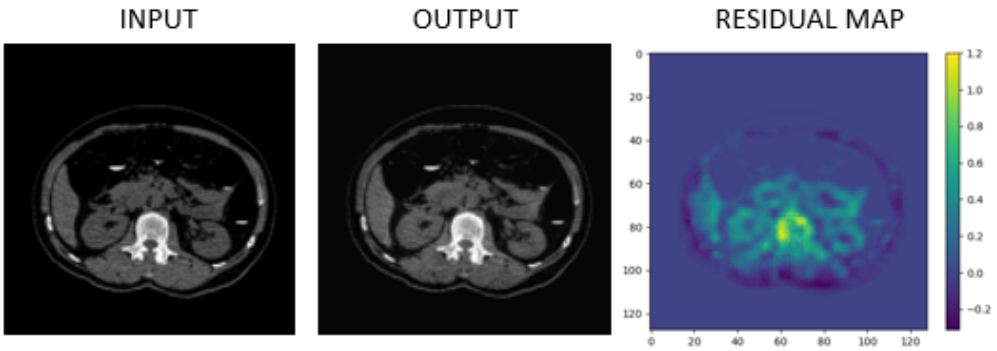


Figure 5.8: *From left to right*: the input image, the output with the added contrast agent and the relative Residual map

Chapter 6

Results and Discussions

In this chapter we describe the results obtained by applying the developed deep generative model.
In the first section we show and comment the qualitative results, highlighting the characteristics that determined their formation.
In the second, and last, section the obtained results will be evaluated from a quantitative point of view, thanks also to the introduction of the Frèchet distance as an evaluation parameter.

6.1 Qualitative results

In the first place, the qualitative results are analysed. The entire project was based on the use of two unpaired sets containing two different types of images. This entails that the results were not evaluated in the classical way, by analysing the data labels and through validation. The generated images that have been obtained, are now presented. These images generated by the CycleGAN are obtained from 6 patients of the Test-Set: where 3 patients present contrast-enhanced CT, and another 3 without contrast. The patients in the Test-Set were divided into four different sections and only the images relating to the renal region were selected, for a total of 180 2D images.
The Figure 6.1 shows some samples of the data generated by the U-Net. In particular, there is the input image that is provided to the generator and the mask that is applied to force the model to modify only the values concerning the abdominal component, avoiding the background. It is also visible the image created by the NN and the relative Residual map. The latter is produced by calculating the difference, between the output and the input data of the generator. This values are then normalized between 0 and 1, where 1 normalized the maximum addition of contrast for the first line while the maximum removal for the second line. In this way, the map makes it easier to identify the portions of the image in which the

model has concentrated and produced the main variations.
In the first line of Figure 6.1, the CycleGAN tries to artificially add contrast: the intensity of the pixels increases mainly in the renal section. Whereas in the second row the generator tries to remove the effect due to the contrast medium: there is the mask and the corresponding Residual map, where it is possible to see that the pixel intensity is lowered but in a less precise way, however, the major changes are in the renal area. To analyze in more detail, the Figure 6.2 shows the application of

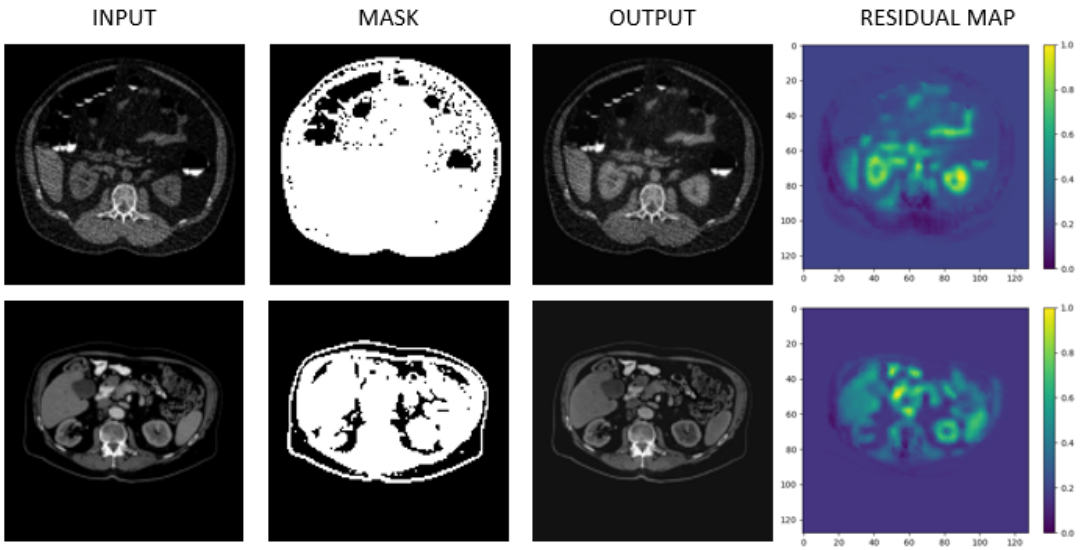


Figure 6.1: *From left to right*: Input image, mask, generated CT image and the relative Residual Map. First line synthetic addition of contrast injection, second line synthetic removal.

the generative model to add the effect of the contrast agent: the input and output images obtained during the translation between the two domains and the corresponding Action Map. Where the kidneys and their blood vessels, which are more distinguishable, have been manually segmented and marked with a red curve. On the other hand, the blue curve indicates the elements of the generated image that are subject to a greater variation with respect to the input data. It is possible to notice that these variations remain inside the abdominal component, thanks to the application of the mask, and involve mainly the renal component. From the output images it can be seen how the net can increase the contrast in the required areas, although in a homogeneous way. It is not able to distinguish between parenchyma and vessels , as happens in a real image with contrast agent. While, Figure 6.3 shows how the generator tries to remove the contrast effect by decreasing the pixel intensity, with a greater variation in the kidneys. The main variations involve the abdominal component, thanks to the application of the binary mask.
The U-Net tries to identify the main features that distinguish data from the two

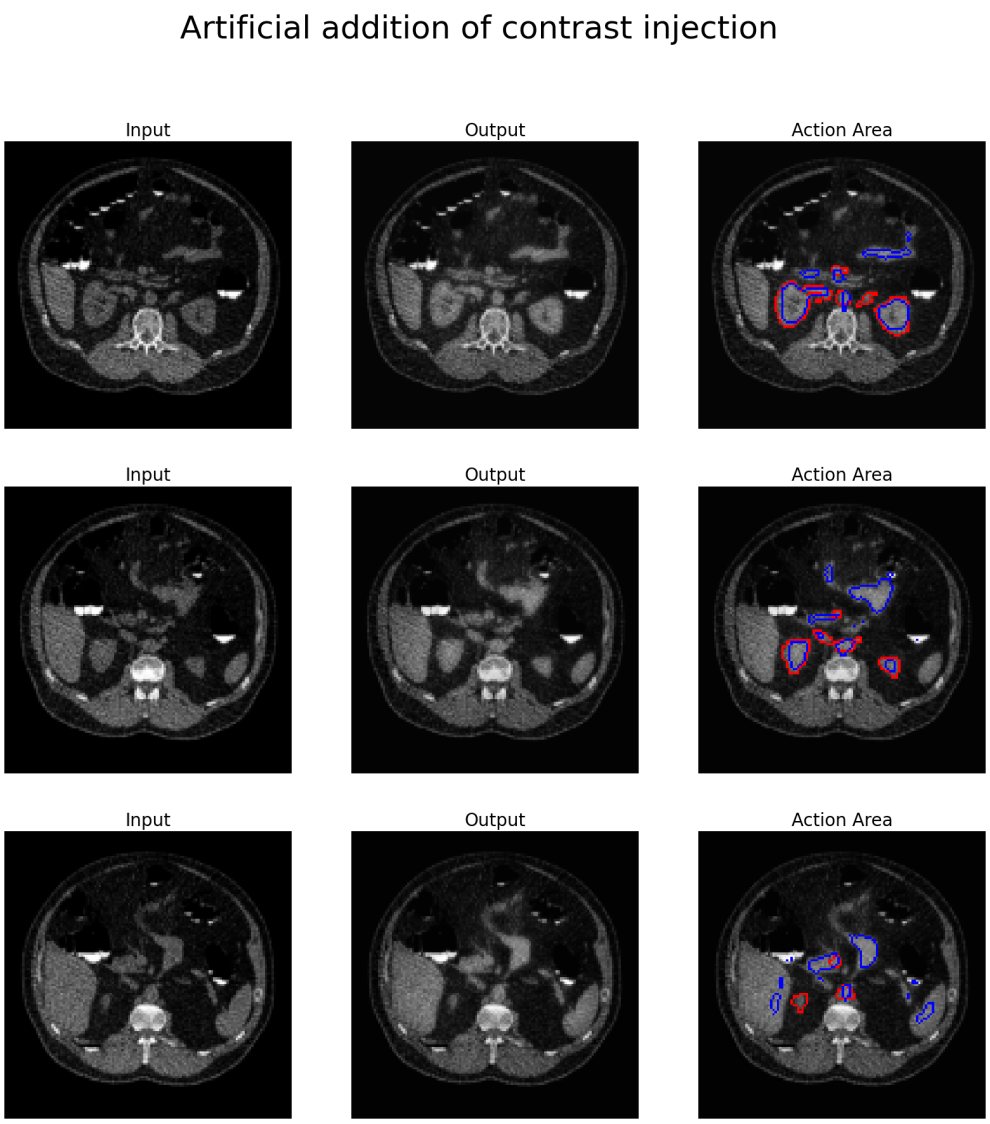


Figure 6.2: *From left to right*: Input image, generated Contrast-enhanced CT image, and the relative Action Map

domains applying distinguishable variations and producing realistic images. From the output images, it is possible to observe how the kidneys and small areas related to the renal arteries are identified and modified by the network, even if only partially. However, the generator leads to the formation of low-contrast images rather than images without contrast where the distribution of pixel intensity is more homogeneous.
From the Res-Net results (Figure 6.4) we can instead see that the NN is able to determine when it is necessary to increase the pixel value to create the contrast

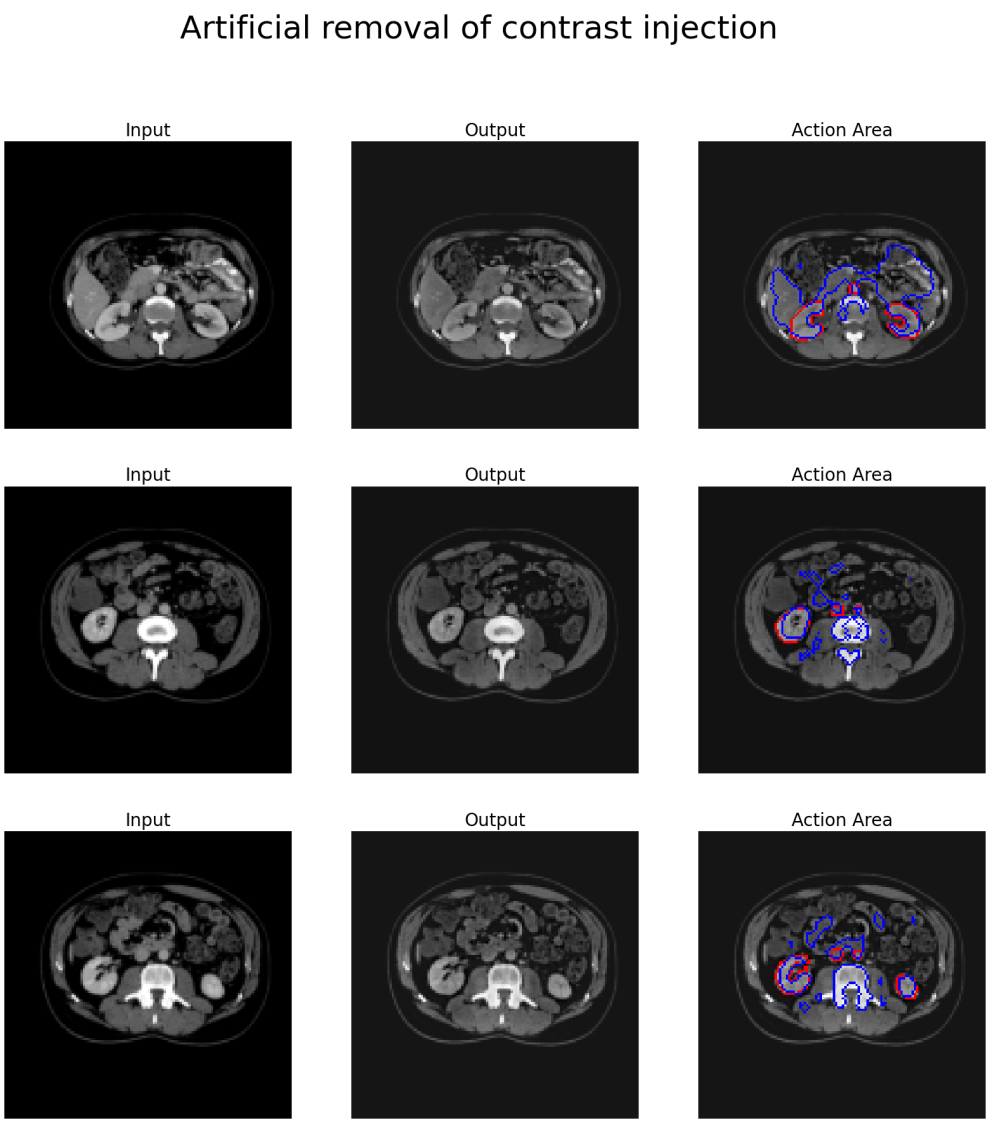


Figure 6.3: *From left to right*: Input image with contrast, generated image, and the relative Action Map

effect and when it was necessary to remove it. However, the modifications involve the entire abdominal component almost equally, with no particular attention to the renal section and the blood vessels. Moreover, these variations are very small and difficult to distinguish from a visual examination.

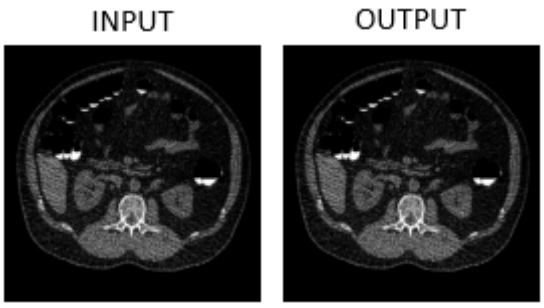


Figure 6.4: Input image without contrast and generated image by Res-NET

6.2 Quantitative results

The results obtained from the generative model are now described from a quantitative point of view. In the first part, the parameters combination identified to improve the stability of the neural networks is described,; then the data distributions obtained are analysed. In the second part, the results obtained are analysed using the Fréchet Inception Distance. The training of the entire CycleGAN model with Res-Net as generators takes about 40 hours to complete. This is due to the simultaneous training of two Res-Net and two discriminators made up of three convolutional layers. In the case where the generators are made up of two U-Net, the training time increases to about 48 hours, due to the larger number of parameters. Different combinations of hyper parameters have been tested for optimal results and an increased stability. The introduction of the Wasserstein distance helps to favour the quality of the generated data and to avoid mode collapse and improve the stability.
As described in the previous chapter, an extremely small learning rate was considered necessary to avoid overfitting. In addition, a batch size of 10 elements is fundamental to guarantee stability to the model, applying batch normalization. A larger size was not possible due to the size of the different graphics cards used. The use of the Adam optimiser and the linear decrease of the learning rate, in the middle of the training phase, favours a better quality of the data generated by the model and avoid instability.
The Figure 6.5 shows the Cycle and Identity Loss trends. It can be observed how the trends, for both the model of deep NNs, tend to converge towards zero, following a stable decreasing trend. However, as described in [33], the use of Cycle Consistency loss facilitates the stabilisation of the generator in the initial steps of the training phase, but becomes an obstacle for the generation of realistic data. This happens mainly in the last stages of the training phase. For this reason, it was decided to gradually decrease the weights of the two losses in the last stages. The same change is applied to the identity loss, to keep the ratio between the two weights constant. The Figure 6.6 shows the variation of the two weights: starting

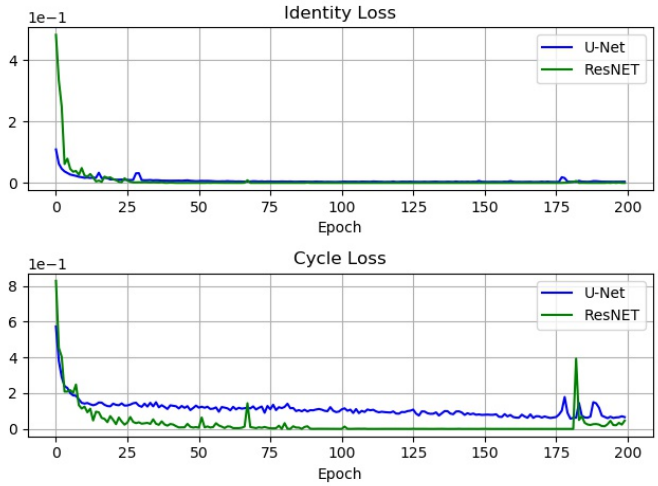


Figure 6.5: Trend of the Cycle and Identity Loss

from the 150th epoch in a gradual way, subtracting a constant K every 10 epochs, until the end of the training phase. For the Cycle loss K is set equal to 1.25, while for the reduction of the Identity loss, K is set at 0.0625. In this way, we decrease the influence of both loss for the generation of new data (without eliminating it completely), while keeping the proportion between them constant.
Moreover one of the main problems related to GANs, is the difficulty of quantitatively evaluating the results obtained, because often it is not sufficient a visual analysis alone. Thus, in order to have a quantitative evaluation of the data generated, two parameters, very common in statistical classification, are calculated: precision and recall.

- Recall is a measure of completeness. It indicates the ratio between the true positives and the sum of true positives and false negatives. It is determined by the ratio between the number of pixels belonging to the two areas (red and blue) and the number of pixels belonging to the red area, (i.e. the portion relating to the kidneys,which is manually segmented). Recall indicates the ability of the generator to correctly identify the portion of the image that have to be subject to variation, corresponding to the renal component and the blood vessels.

- Precision is a measure used for classification or pattern recognition. It indicates the ratio between the number of true positives and the sum of true positives and false positives. It is determined by the ratio between the sum of the value of the pixels of the residual map, belonging to the two areas visible in the Action map (red and blue), and the total sum of the value of the pixels in the blue area. . Precision makes it possible to determine if most of the changes in the generator have been concentrated in the target portion of each image.

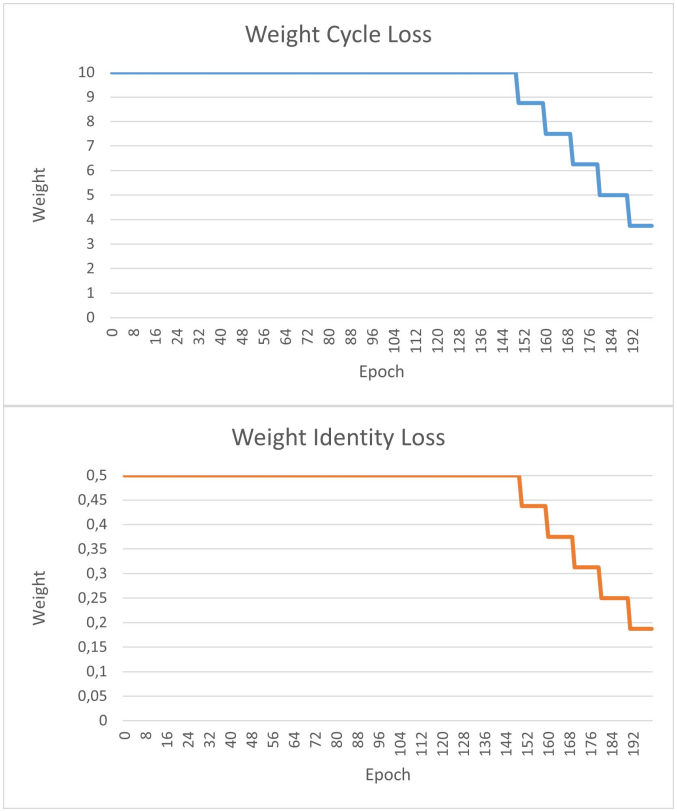


Figure 6.6: Evolution of the weights of Cycle *(upper)* and Identity *(lower)* loss

For our study, when the the combination of hyper-parameters providing the best results has been determined, 3 patients, from each domain (belonging to the Test-Set), are chosen to generate 10 new images, by means of the CycleGAN. In this way, 20 fake images are obtained, completely generated by the U-Net. The first three images for each domain are visible in the Figure 6.2 and 6.3, the others are included in the appendix. On these 20 fake images, the Recall of the action area and the Precision of intensity variation, are applied. The figure 6.7 shows results on data of syntethic contrast enhanced images obtained from CT Colonography (non-contrast) database. It can be seen that the precision is often higher than the recall value. This indicates that the model does not completely identify the section of the image that needs to be modified, but the parts that undergo greater variation belong to the renal component. This reflects what can be seen from the residual map in the Figure 6.2. Furthermore, in some images, like in the third row of the Figure 6.2, the kidneys are not much present in the CT image, but only a portion with also a small part of the renal artery is visible. For this reason,the model has more difficulty to identify this small anatomical portion.
The Figure 6.8 shows instead results on the data of synthetic non-contrast images obtained from the Pancreas CT (contrast) database. In this case the recall values

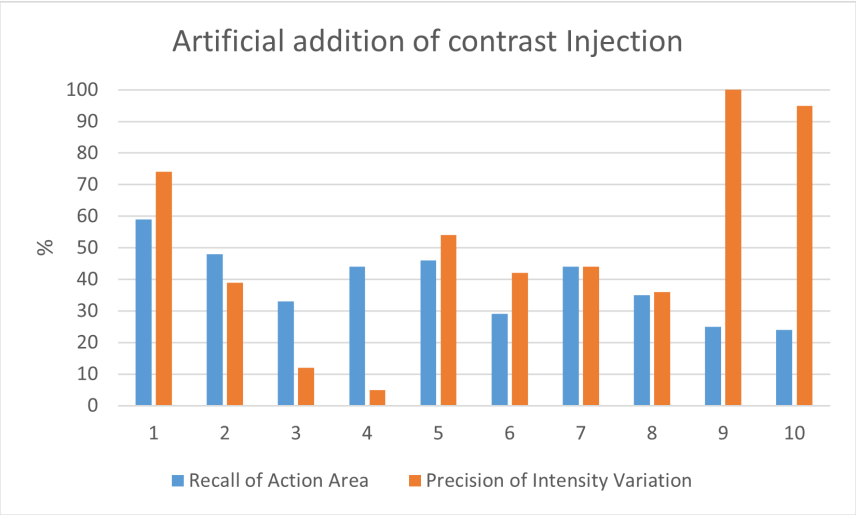


Figure 6.7: Recall and Precision of 10 images generated by the U-Net to add the contrast effect

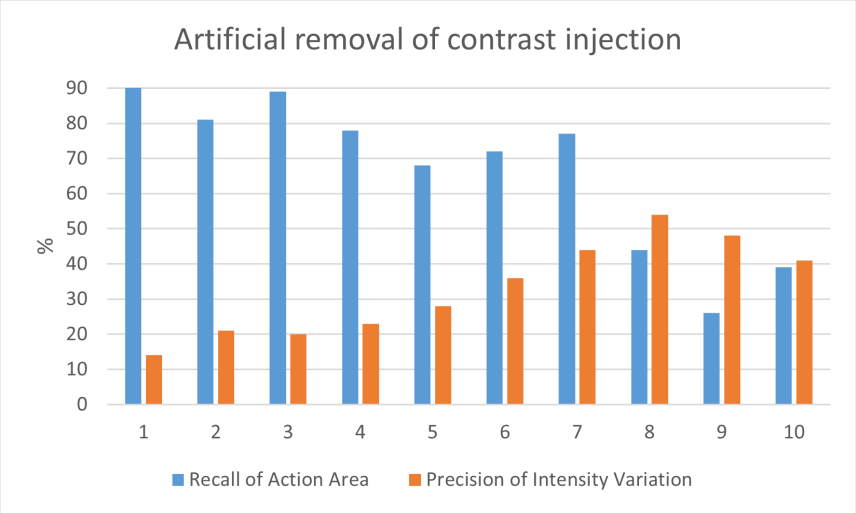


Figure 6.8: Recall and Precision of 10 images generated by the U-Net to remove the contrast effect artificially

are much higher than the precision. This implies that the model modifies several elements of the image, which should not be affected by the contrast changes, but among these elements there are also the target renal and vessels areas. These results reflect the qualitative ones, where we can see that it is as if we switched to a low-contrast rather than a non-contrast, as mentioned before. On reason of this behaviour could be because the contrast medium in the arterial phase does not only involve the kidneys and vessel, but can also cause an effect on other anatomical elements. For this reason, the generator modify elements that do not belong to

the interested area. Anyway, it is still unclear how this problem arises only in this "direction" of change.
It is important to point out that the results obtained up to now have been calculated using the blue action are, calculated by applying a threshold of 0.7 (with a maximum of 1.0) on the residual map. In fact this was the result of a pixel-by-pixel difference between the output image and the input. The figure 6.10 shows how the precision in the two translations changes according to the increase of the threshold (from 0.7 to 0.8 and 0.9). When the generator tries to remove the contrast, it is observed that the precision increases. This indicates that the major changes actually appears inside the renal component for the most part of the test images. When the generator tries to add the effect of contrast, as the threshold is increased, the precision remains the same or gets worse. This indicates that the generator is not only applying the main changes within the renal component, but other anatomical portions are affected. In the Figure 6.9, the precision is calculated on a test where the binary mask was not applied during the training phase. It can be seen that the precision (whit the same value of threshold of 0.9) when the mask is not applied decrease a lot. The same happens in the opposite direction, when the generator, tries to insert the contrast effect. This indicates that the binary mask is essential for both generators to focus on the abdominal component, ignoring the background and black holes present within the anatomical components.

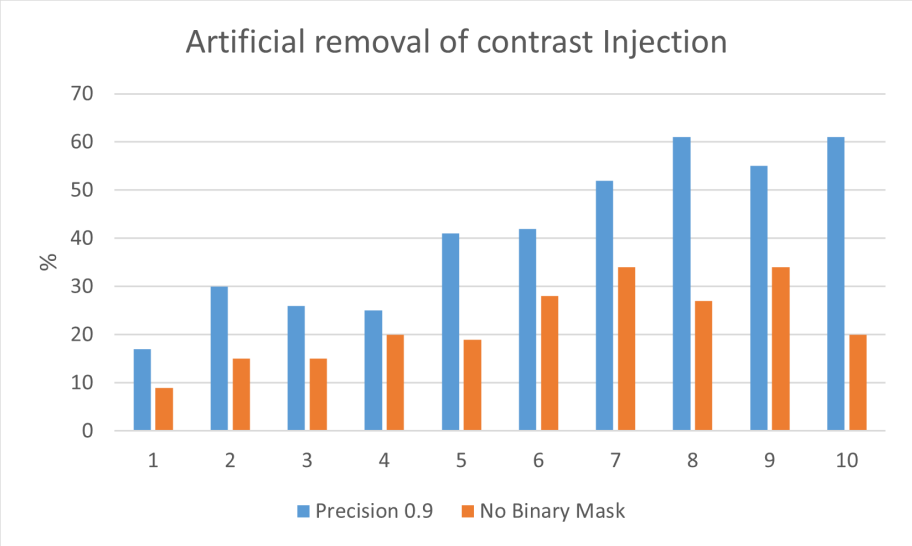


Figure 6.9: Variation of precision when the binary mask is applied

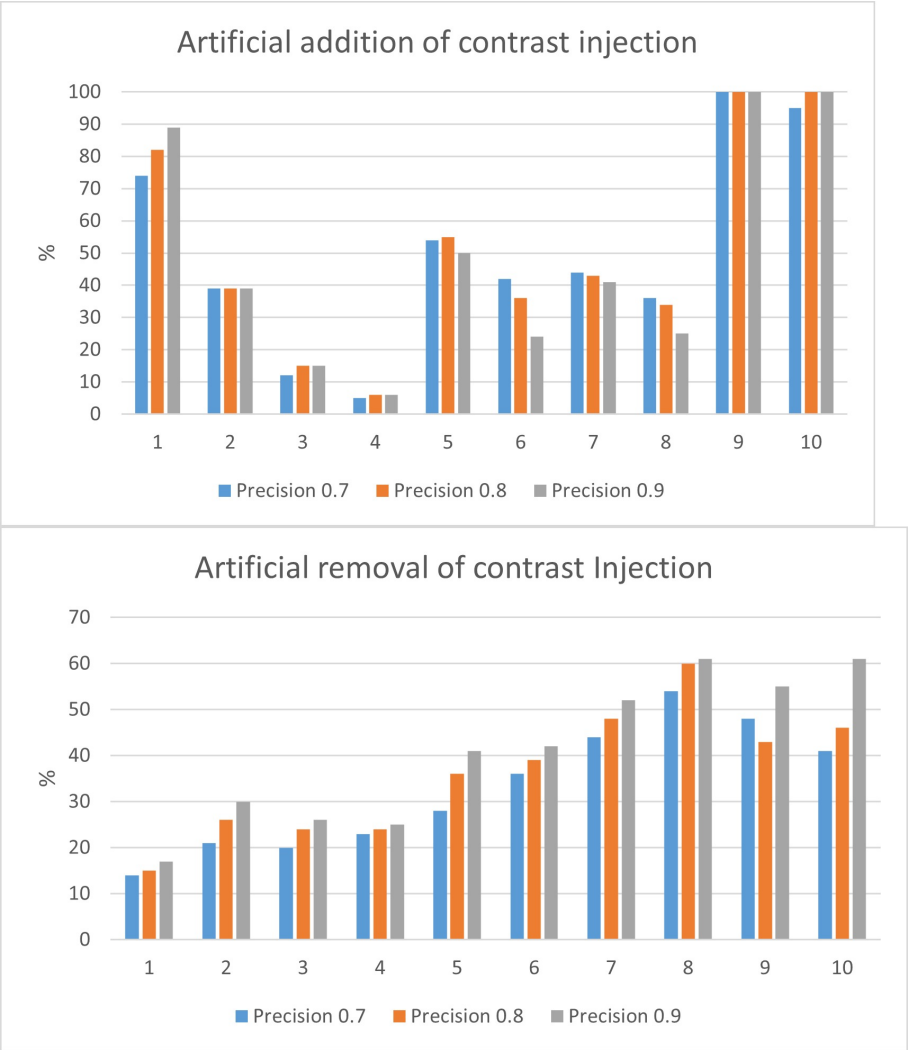


Figure 6.10: Variation of precision according to threshold value

6.2.1 The application of Fréchet Inception Distance

As stated above, one of the main problems related to generative models is the difficulty of qualitatively evaluating the results obtained. In fact, the evaluation of loss and validation are not always directly related to the quality of the data obtained by the generator. For this reason, models are very often evaluated through the implementation of an additional NN, such as a classifier or segmentor, which are applied directly to the new generated images. In this way, it can be assessed the improvements and the performance of this network, with the introduction of new data. Examples of these applications can be seen in the third chapter. New parameters have been introduced for the objective evaluation of generator performance, directly applicable to the new data. To do this, the Fréchet Inception Distance

(FID) was applied as a comparative instrument. The FID score, also called as Wasserstein-2 distance, has lately become more and more popular, replacing the inception score (ID score). Indeed, ID Score only considers the generated images, without comparing them to real data. It also necessitates a classification Net, trained on ImageNet. The FID score measures the distance between two normal distributions, by measuring the mean and the standard deviation [60].
In the case of multivariate distributions, like those we have obtained from CycleGAN, the covariance matrix is calculated in the formula, which allows us to determine the relationship between the two distribution of image data that have been obtained.

$$\text{FID} = ||\mu_r - \mu_g||^2 + \text{Tr}(\Sigma_r + \Sigma_g - 2(\Sigma_r\Sigma_g)^{1/2}), \tag{6.1}$$

Where μ_g and μ_r are the mean of the generated and real and images and Σ_g Σ_r are the relative covariance matrix. Tr represents the trace linear algebra operation, e.g. the sum of the elements along the main diagonal of the square matrix. The formula is applied on two different data distributions, which correspond to the data belonging to the real dataset and the data obtained from the generator. A low FID score indicates a small distances between the two domain of images; which indicates an high quality of the data distribution obtained by the model.
To calculate this parameter we use a network model InceptionV3, which is an image classifier, trained with a large number of images taken from ImageNet.
In particular, to calculate the FID score, it is necessary to cut at the level of the third and last pooling layer, before the output classification layer. The pooling layer of the model is used to capture the specific features of an input image. The last pooling layer consists of 2,048 activation; each image from both domains is converted as 2,048 activation features. This process is known as the coding vector (or feature vector) for the image. In this way a vector with 2,048 features is obtained for the collection composed of the images generated by the generator. The feature vector is then also calculated from the real images to provide a reference of the distribution of the original data. In this way, the final result will be the set of 2.048 feature vectors for real and fake images, and the Equation 6.1 is applied to these vectors.
However, it is necessary to point out that there are some limitations regarding the use of the FID score, for this reasons it is not yet always used. This is because it requires a pre-trained model on a large number of images whose characteristics may not be helpful in identifying the features of the analysed data. As in our case, where the ImageNet database is different from the medical images databases used. It should also be independent of the number of samples that are included, however, due to the presence of bias, this can affect the value of the metric. Finally, it uses limited statistical data: mean and variance are in fact first order moments. Therefore they are not necessarily able to cover all aspects of the data distribution, which moreover do not present, a perfect multivariate normal distribution. For this

reason, qualitative evaluation is still essential to identify the desired model and to debug it.
The Test-Set consists of 5 patients, from each point domain 256 fake images are taken. On these data the FID score is calculated, to determine the distance between 2 sets of images. As a reference parameter we take the distance between 2 sets of images, composed of 256 images taken from CT Pancreas, belonging to domain A (with contrast injected), and CT Colonography, belonging to domain B (contrast free). The Table 6.1 shows the results of the Frechet distance, where A and B indicate the corresponding domain.

- The distance between the two sets of real images is 0.10. The small value indicates that the distance between the two sets of images is very close. This is because, the presence of contrast does not change physiological aspects of the patient, but only the intensity of some pixels. Furthermore, for all data, part of the image is composed of the background.

Model	A fake - B fake	A fake - A real	B fake - B real
U-Net (no Mask)	0.07	0.62	0.64
ResNET	0.08	0.40	0.50
U-Net	0.3	0.23	0.50

Table 6.1: FID score for different models

The first column evaluates the distance between the two sets of images completely generated by the generator (A fake - B fake). In this way we evaluate the difference between the two types of images created based on the relationship to the reference distance between real images. In the first case analyzed the two false distributions turned out to be very similar to each other. Moreover it can be seen, in the second and third columns that, not using the binary mask leads to the generation of a data distribution that is very different from the real one, for this reason the FID score is high.
The ResNET, however, is not able to modify much the data provided as input. This is also indicated by the FID score which is very close to the reference value. For this reason, the generated images are more distant, and therefore more different, than the corresponding real images taken from the two databases.
The U-Net shows that it generates fake images with contrast very similar to the real ones (A fake - A real); in the opposite direction instead (B fake - B real) the Frechet distance indicates that the results are not yet optimal. However, the U-Net also shows a high ability to differentiate the generated images, belonging to the two domains (A fake - B fake). For this reason, the distance between the two sets of fake images is three times higher than the reference distance between the real distributions.

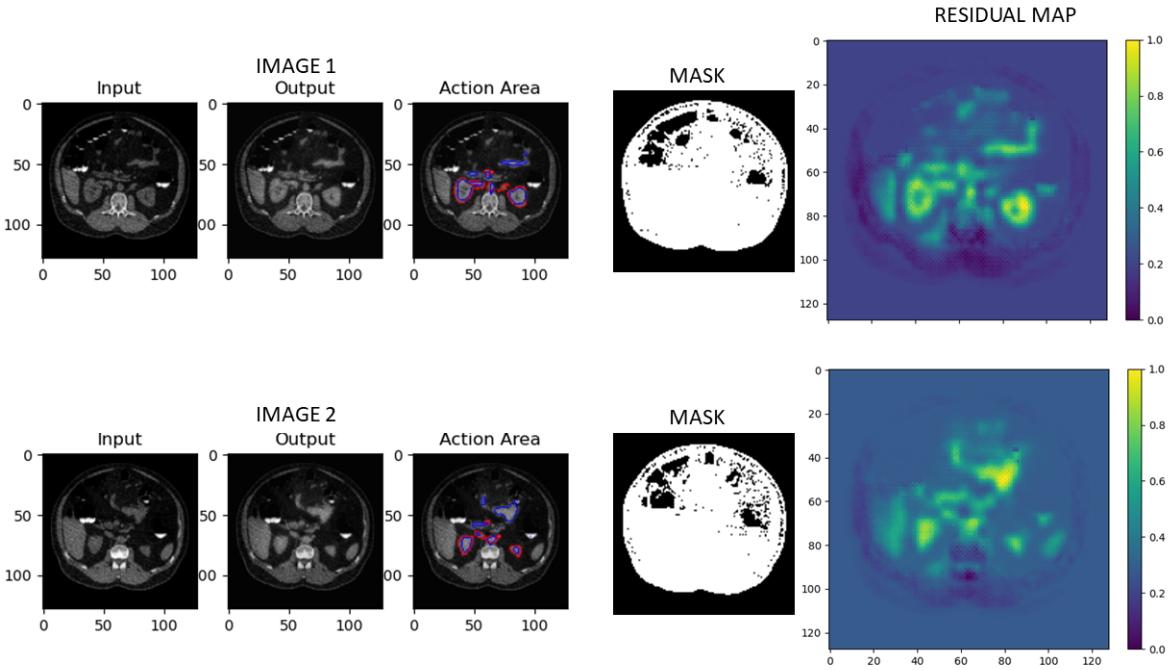
6.3 Conclusion and Future steps

The objective of this thesis was to propose a method to perform a cross-domain CT image translation, through a specific generative model, the CycleGAN. In particular, we tried to obtain from a set of unpaired CT images with contrast agents, acquired in the arterial phase, CT images where contrast injection is not present and vice versa.
The project starts from the original CycleGAN implementation and then we apply a series of modifications to improve the quality of the output data distribution. In the first part, we focused on identifying the most appropriate type of medical images for this type of study, selecting the most suitable data. It is understood that this selection and pre-processing of data is fundamental to improve the quality of the data used an input and to reduce their variability. In fact, to improve the quality of the results it was necessary to identify the portion relating to the kidneys and main renal vessels within each full scan of the training patients. In this way we exclude those anatomical components, such as the lungs or femoral bones, which are not involved in the insertion of the contrast medium, leading the network to focus on unnecessary elements. For the same reason we only limited the use to healthy patients, to avoid possible distortions. In fact, if the generator tried to add contrast, analysing pathological patients, it could create hallucinations or distortions because of the lack of information on images without contrast, as a results of the homogeneity of the different structures within it. In addition, the application of the binary mask for each input image has been shown to be fundamental in allowing the NN to focus on the anatomical components and to avoid modifying the intensity of the background pixels or the spaces between organs.
The proposed method makes it possible to obtain additional images with different characteristics, from some specific medical image databases. In our case the variation is related to the insertion of the contrast agent. Through this technique way it is possible to make a first step towards solving one of the main problems related to available medical databases: the lack of a large amount of data available. The work carried out does not aim to prevent patients from inserting contrast medium during medical examinations, but instead aims to increase the amount of data available for deep networks (for data augmentation, for example). Although the results are not yet very accurate. It is observed that the model was able to identify when it was necessary to increase, or decrease the pixel density , in order to perform the translation from one domain to another. Furthermore it also identifies the portions of the kidneys and the major vessels subject to variation when it has to introduce contrast agent, but not completely or very accurately. There has been a significant improvement in image quality compared to the first experiments with the original CycleGAN model. However, the generative model is still not able to obtain contrast-free images, instead it produces low contrast CT images.
As described in the chapter on the State of the Art there is no standard procedure

to evaluate the performance of the GANs. In the results section, an attempt is made to evaluate the performance of CycleGAN from a quantitative point of view, using statistical parameters to assess the accuracy of the generated data distributions. Furthermore, the Fréchet inception distance is used as a measure to evaluate the distance between the real data distributions and those generated by the Neural Network. This score is becoming popular in this field. In our project, it was applied to evaluate the conditions for obtaining the most realistic data possible. Its use can play an important role in the comparison and in the evaluation of the performance of different GANs developed in future research.
The generative model can be improved, overcoming one of the main problems of this study, the limited variability of the data. It is possible to take a step forward by also using pathological patients, such as thrombosis. In this way, it can be analysed if the generative model is able, based on the information in the image, to create the corresponding contrast-enhanced image. Indeed in future work we might be able to study if the generative model can detect the information relative to pathological elements in a contrast-free image, while the doctor is unable to extrapolate this kind of information. A further improvement that can be applied is the addition of an independent and trained classifier net to identify and classify the images used as input according to their anatomical components present within. This makes the algorithm independent of the chosen database and provides a more precise division of the images than the four section division applied in the training phase.
Moreover, by adding an additional local discriminator that analyses patches taken from the image to focus locally on specific regions of the image, it is possible to make a further improvement that involves the discriminator

Appendices

Artificial Addition of Contrast Injection



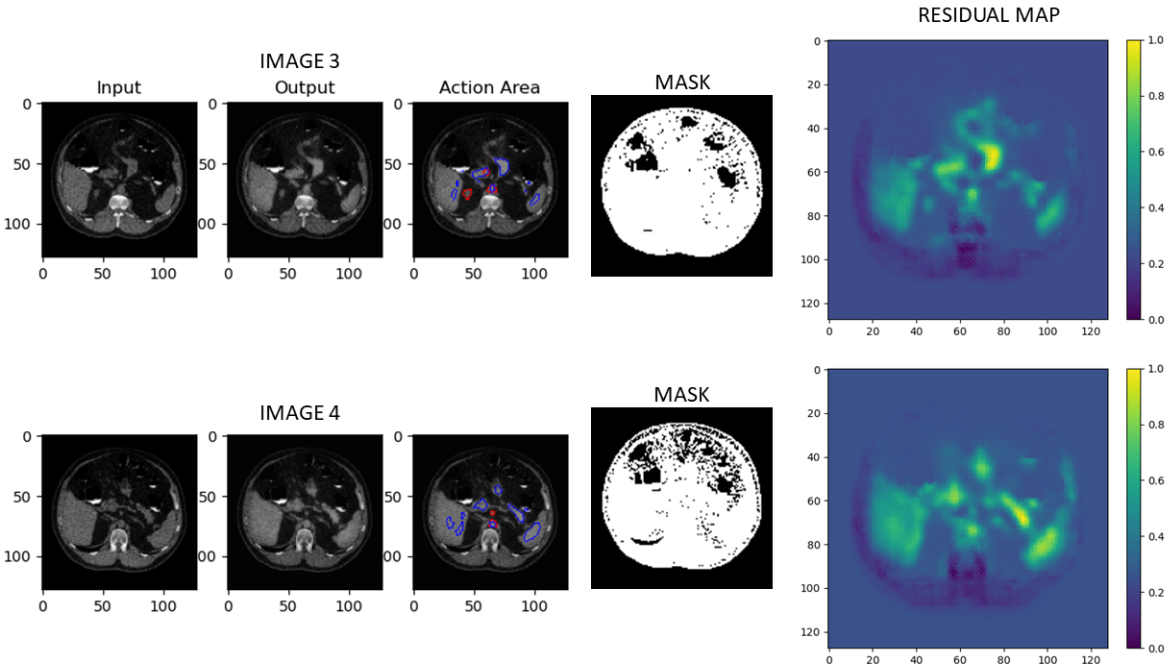


IMAGE 3
Input
Output
Action Area
MASK
RESIDUAL MAP
IMAGE 4
MASK

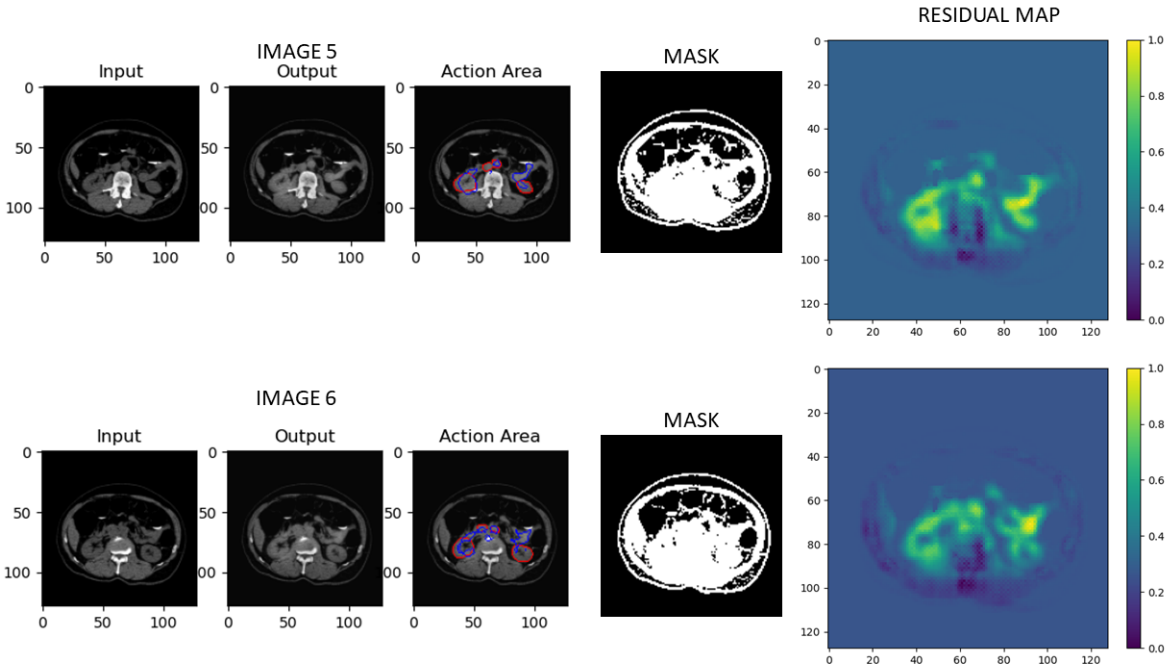


IMAGE 5
Input
Output
Action Area
MASK
RESIDUAL MAP
IMAGE 6
MASK

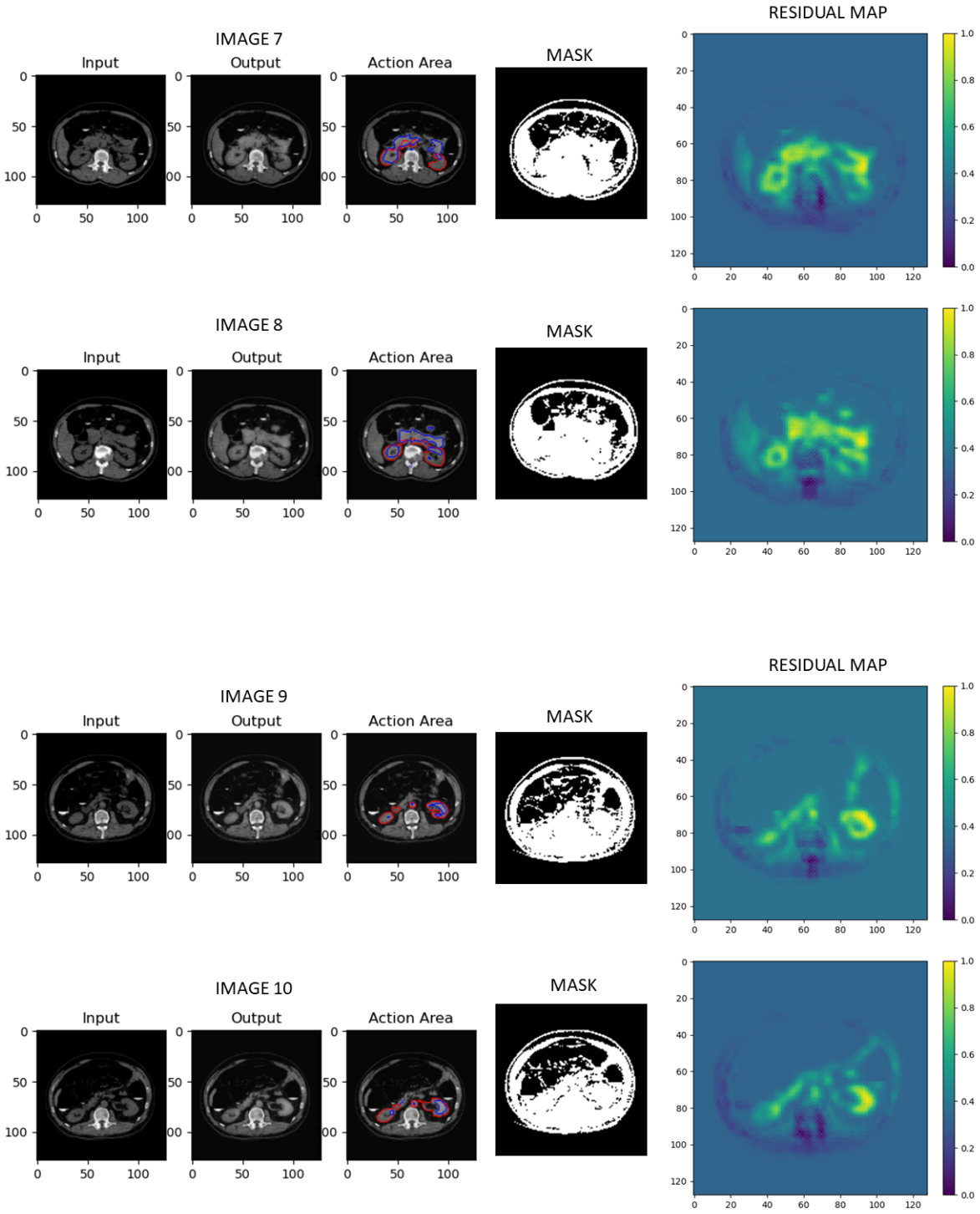
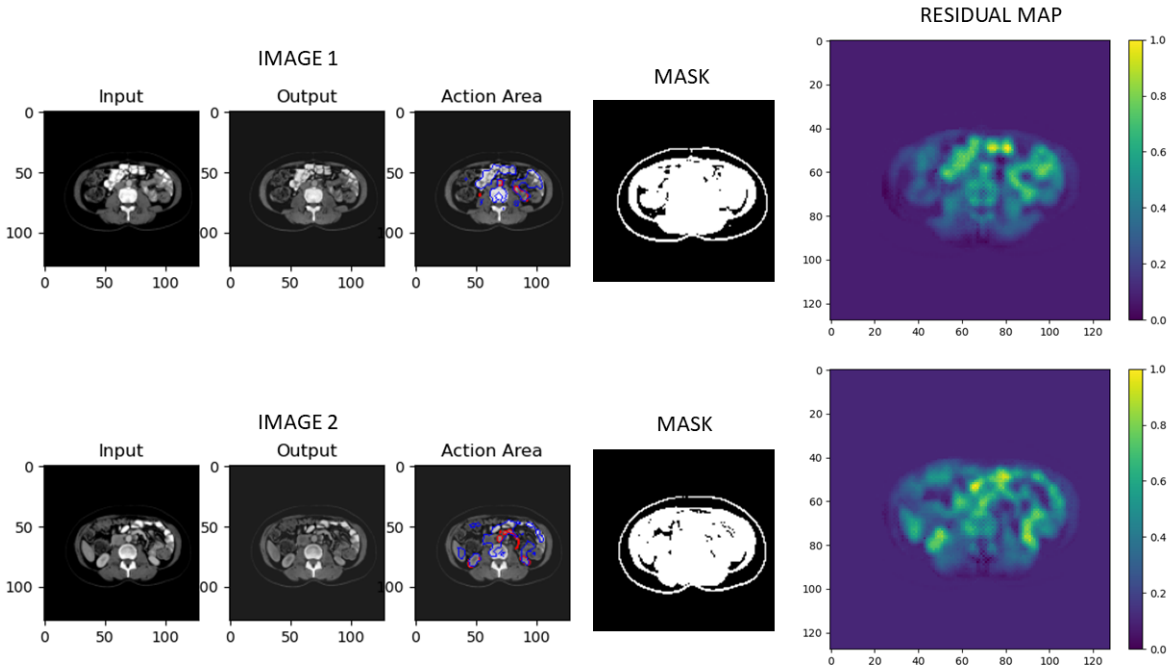


IMAGE 7
Input
Output
Action Area
MASK
RESIDUAL MAP
IMAGE 8
Input
Output
Action Area
MASK
IMAGE 9
Input
Output
Action Area
MASK
RESIDUAL MAP
IMAGE 10
Input
Output
Action Area
MASK

Artificial Removal of Contrast Injection



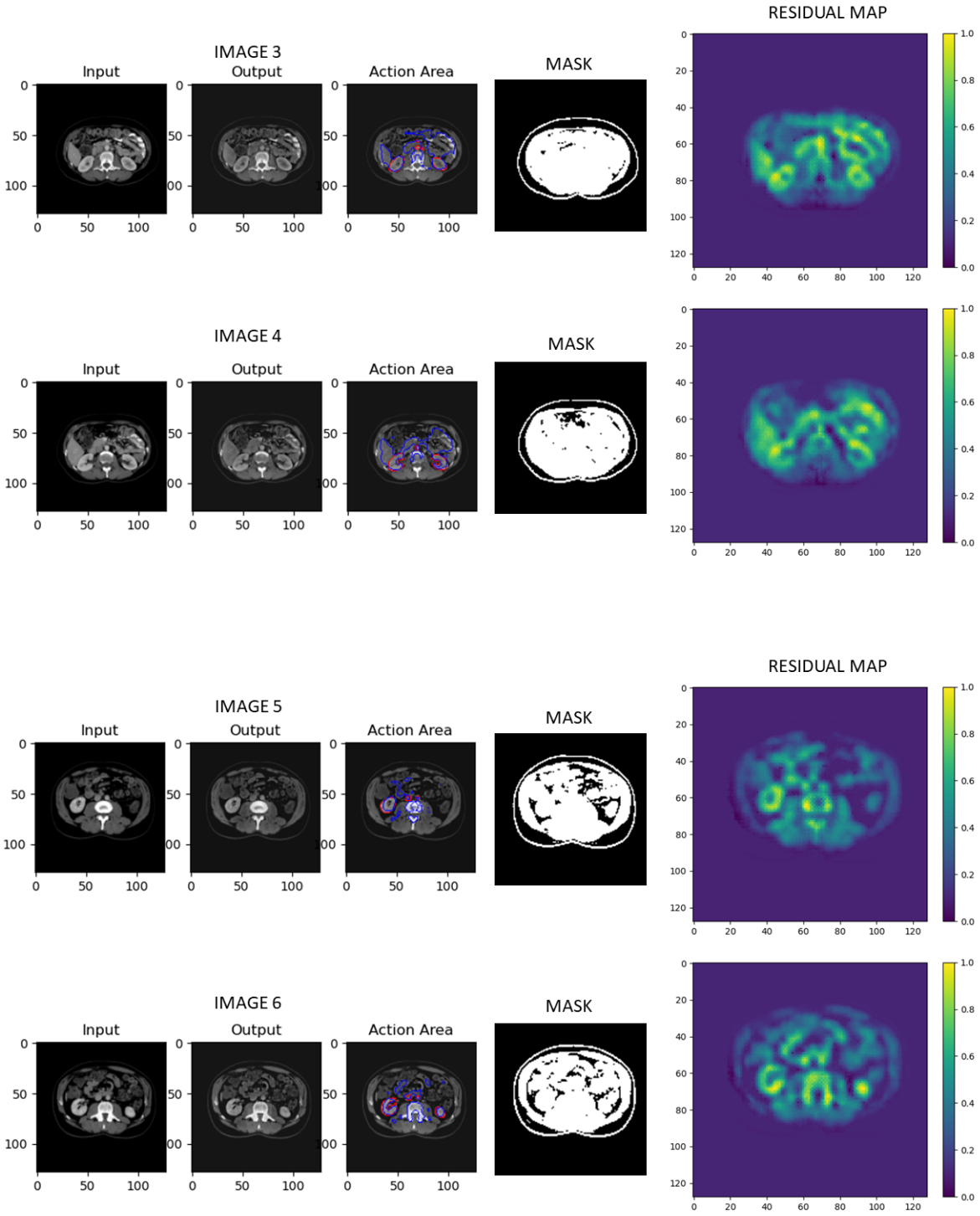
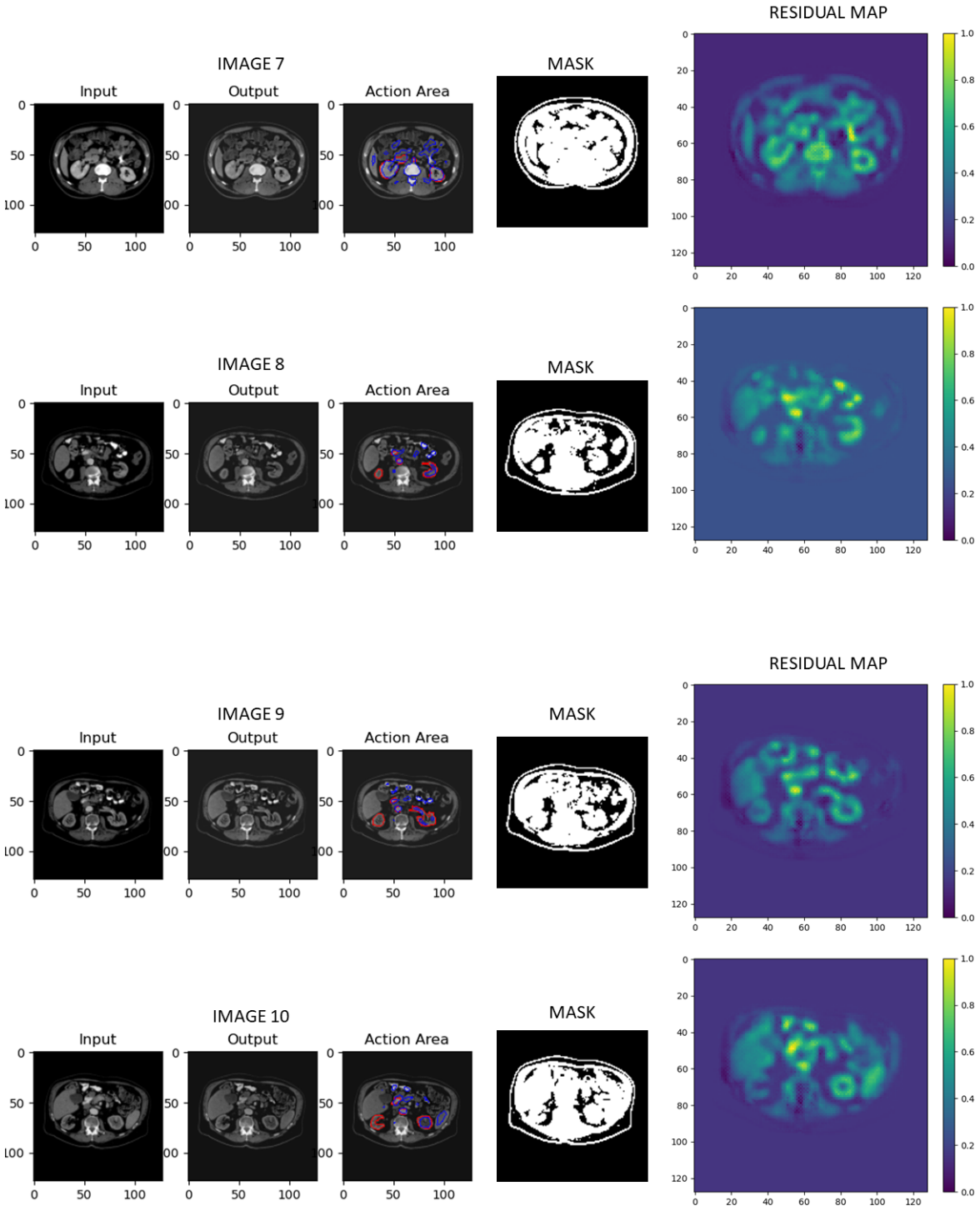


IMAGE 3
Input
Output
Action Area
MASK
RESIDUAL MAP
IMAGE 4
Input
Output
Action Area
MASK
IMAGE 5
Input
Output
Action Area
MASK
RESIDUAL MAP
IMAGE 6
Input
Output
Action Area
MASK


RESIDUAL MAP
IMAGE 7
Input
Output
Action Area
MASK
IMAGE 8
Input
Output
Action Area
MASK
RESIDUAL MAP
IMAGE 9
Input
Output
Action Area
MASK
IMAGE 10
Input
Output
Action Area
MASK

References

[1] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, pp. 2223–2232, 2017.

[2] F. Giovagnorio, *Manuale di Diagnostica per Immagini nella Pratica Medica*. Esculapio, 2017.

[3] W. R. Hendee and E. R. Ritenour, *Medical imaging physics*. John Wiley & Sons, 2003.

[4] M. Chappell, *Principles of Medical Imaging for Engineers*. Springer, 2019.

[5] S. Melisa and S. Vikas, *Radiology Basics: Cross-sectional Imaging*. Melisa Sia, 2015.

[6] H. Lusic and M. W. Grinstaff, "X-ray-computed tomography contrast agents," *Chemical reviews*, vol. 113, no. 3, pp. 1641–1666, 2013.

[7] B. M. Yeh, P. F. FitzGerald, P. M. Edic, J. W. Lambert, R. E. Colborn, M. E. Marino, P. M. Evans, J. C. Roberts, Z. J. Wang, M. J. Wong, *et al.*, "Opportunities for new ct contrast agents to maximize the diagnostic potential of emerging spectral ct technologies," *Advanced drug delivery reviews*, vol. 113, pp. 201–222, 2017.

[8] J. Y. Chin, E. Goldstraw, P. Lunniss, and K. Patel, "Evaluation of the utility of abdominal ct scans in the diagnosis, management, outcome and information given at discharge of patients with non-traumatic acute abdominal pain," *The British journal of radiology*, vol. 85, no. 1017, pp. e596–e602, 2012.

[9] S. Sheth and E. K. Fishman, "Multi-detector row ct of the kidneys and urinary tract: techniques and applications in the diagnosis of benign diseases," *Radiographics*, vol. 24, no. 2, pp. e20–e20, 2004.

[10] R. Mathew, G. Kaveriappa, M. Shetty, and H. Suresh, "Subcutaneous urinoma: A rare sequelae to percutaneous nephrolithotomy," *Muller Journal of Medical Sciences and Research*, vol. 6, no. 1, pp. 78–80, 2015.

[11] D. Foster, *Generative Deep Learning*. O'Reilly Media, Inc, 2019.

[12] M. A. Nielsen, *Neural networks and deep learning*, vol. 2018. Determination press San Francisco, CA, 2015.

[13] R. Reed and R. J. MarksII, *Neural smithing: supervised learning in feedforward artificial neural networks*. Mit Press, 1999.

[14] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep learning*, vol. 1. MIT press Cambridge, 2016.

[15] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[16] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015.

[17] N. Ketkar and E. Santana, *Deep Learning with Python*, vol. 1. Springer, 2017.

[18] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[19] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[20] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241, Springer, 2015.

[21] Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger, "3d u-net: learning dense volumetric segmentation from sparse annotation," in *International conference on medical image computing and computer-assisted intervention*, pp. 424–432, Springer, 2016.

[22] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-net: Fully convolutional neural networks for volumetric medical image segmentation," in *2016 fourth international conference on 3D vision (3DV)*, pp. 565–571, IEEE, 2016.

[23] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

[24] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1–9, 2015.

[25] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," 2014.

[26] A. Odena, C. Olah, and J. Shlens, "Conditional image synthesis with auxiliary classifier gans," in *International conference on machine learning*, pp. 2642–2651, PMLR, 2017.

[27] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein gan," *arXiv preprint arXiv:1701.07875*, 2017.

[28] S. Nowozin, B. Cseke, and R. Tomioka, "f-gan: Training generative neural samplers using variational divergence minimization," in *Advances in neural information processing systems*, pp. 271–279, 2016.

[29] A. B. L. Larsen, S. K. Sønderby, H. Larochelle, and O. Winther, "Autoencoding beyond pixels using a learned similarity metric," in *International conference on machine learning*, pp. 1558–1566, PMLR, 2016.

[30] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv:1411.1784*, 2014.

[31] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1125–1134, 2017.

[32] M.-Y. Liu, T. Breuel, and J. Kautz, "Unsupervised image-to-image translation networks," in *Advances in neural information processing systems*, pp. 700–708, 2017.

[33] T. Wang and Y. Lin, "Cyclegan with better cycles," 2019. Available at `https://ssnl.github.io/better_cycles/report.pdf`.

[34] A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, and R. Webb, "Learning from simulated and unsupervised images through adversarial training," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2107–2116, 2017.

[35] X. Yi, E. Walia, and P. Babyn, "Generative adversarial network in medical imaging: A review," *Medical image analysis*, vol. 58, p. 101552, 2019.

[36] M. J. Chuquicusma, S. Hussein, J. Burt, and U. Bagci, "How to fool radiologists with generative adversarial networks? a visual turing test for lung cancer diagnosis," in *2018 IEEE 15th international symposium on biomedical imaging (ISBI 2018)*, pp. 240–244, IEEE, 2018.

[37] M. Frid-Adar, I. Diamant, E. Klang, M. Amitai, J. Goldberger, and H. Greenspan, "Gan-based synthetic medical image augmentation for increased cnn performance in liver lesion classification," *Neurocomputing*, vol. 321, pp. 321–331, 2018.

[38] D. Korkinof, T. Rijken, M. O'Neill, J. Yearsley, H. Harvey, and B. Glocker, "High-resolution mammogram synthesis using progressive generative adversarial networks," *arXiv preprint arXiv:1807.03401*, 2018.

[39] E. Kang, H. J. Koo, D. H. Yang, J. B. Seo, and J. C. Ye, "Cycle-consistent adversarial denoising network for multiphase coronary ct angiography," *Medical physics*, vol. 46, no. 2, pp. 550–562, 2019.

[40] C. You, G. Li, Y. Zhang, X. Zhang, H. Shan, M. Li, S. Ju, Z. Zhao, Z. Zhang, W. Cong, *et al.*, "Ct super-resolution gan constrained by the identical, residual, and cycle learning ensemble (gan-circle)," *IEEE transactions on medical imaging*, vol. 39, no. 1, pp. 188–203, 2019.

[41] C.-B. Jin, H. Kim, M. Liu, W. Jung, S. Joo, E. Park, Y. S. Ahn, I. H. Han, J. I. Lee, and X. Cui, "Deep ct to mr synthesis using paired and unpaired data," *Sensors*, vol. 19, no. 10, p. 2361, 2019.

[42] J. M. Wolterink, A. M. Dinkla, M. H. Savenije, P. R. Seevinck, C. A. van den

Berg, and I. Išgum, "Deep mr to ct synthesis using unpaired data," in *International workshop on simulation and synthesis in medical imaging*, pp. 14–23, Springer, 2017.

[43] Z. Zhang, L. Yang, and Y. Zheng, "Translating and segmenting multimodal medical volumes with cycle-and shape-consistency generative adversarial network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 9242–9251, 2018.

[44] Y. Pan, M. Liu, C. Lian, T. Zhou, Y. Xia, and D. Shen, "Synthesizing missing pet from mri with cycle-consistent generative adversarial networks for alzheimer's disease diagnosis," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 455–463, Springer, 2018.

[45] S. U. Dar, M. Yurt, L. Karacan, A. Erdem, E. Erdem, and T. Çukur, "Image synthesis in multi-contrast mri with conditional generative adversarial networks," *IEEE transactions on medical imaging*, vol. 38, no. 10, pp. 2375–2388, 2019.

[46] P. Welander, S. Karlsson, and A. Eklund, "Generative adversarial networks for image-to-image translation on multi-contrast mr images-a comparison of cyclegan and unit," *arXiv preprint arXiv:1806.07777*, 2018.

[47] J. T. Guibas, T. S. Virdi, and P. S. Li, "Synthetic medical images from dual generative adversarial networks," *arXiv preprint arXiv:1709.01872*, 2017.

[48] V. Sandfort, K. Yan, P. J. Pickhardt, and R. M. Summers, "Data augmentation using generative adversarial networks (cyclegan) to improve generalizability in ct segmentation tasks," *Scientific reports*, vol. 9, no. 1, pp. 1–9, 2019.

[49] S. Kaji and S. Kida, "Overview of image-to-image translation by use of deep neural networks: denoising, super-resolution, modality conversion, and reconstruction in medical imaging," *Radiological physics and technology*, vol. 12, no. 3, pp. 235–248, 2019.

[50] Y. Hiasa, Y. Otake, M. Takao, T. Matsuoka, K. Takashima, A. Carass, J. L. Prince, N. Sugano, and Y. Sato, "Cross-modality image synthesis from unpaired data using cyclegan," in *International workshop on simulation and synthesis in medical imaging*, pp. 31–41, Springer, 2018.

[51] S. Kida, S. Kaji, K. Nawa, T. Imae, T. Nakamoto, S. Ozaki, T. Ohta, Y. Nozawa, and K. Nakagawa, "Cone-beam ct to planning ct synthesis using generative adversarial networks," *arXiv preprint arXiv:1901.05773*, 2019.

[52] Y. Zhu, Y. Tang, Y. Tang, D. C. Elton, S. Lee, P. J. Pickhardt, and R. M. Summers, "Cross-domain medical image translation by shared latent gaussian mixture model," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 379–389, Springer, 2020.

[53] J. P. Cohen, M. Luck, and S. Honari, "Distribution matching losses can hallucinate features in medical image translation," in *International conference on medical image computing and computer-assisted intervention*, pp. 529–536, Springer, 2018.

[54] B. Whitcher, V. J. Schmid, and A. Thornton, "Working with the dicom and nifti data standards in r," *Journal of Statistical Software*, no. 6, 2011.

[55] B. Belucci Teixeira, *Deep Learning Approach to Kidney and Kidney Tumor Semantic Segmentation.* Final study project, Telecom Paris, 2020.

[56] J. A. Parker, R. V. Kenyon, and D. E. Troxel, "Comparison of interpolating methods for image resampling," *IEEE Transactions on medical imaging*, vol. 2, no. 1, pp. 31–39, 1983.

[57] I. Goodfellow, "Nips 2016 tutorial: Generative adversarial networks," *arXiv preprint arXiv:1701.00160*, 2016.

[58] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of wasserstein gans," in *Advances in neural information processing systems*, pp. 5767–5777, 2017.

[59] N. U. Din, K. Javed, S. Bae, and J. Yi, "A novel gan-based network for unmasking of masked face," *IEEE Access*, vol. 8, pp. 44276–44287, 2020.

[60] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," in *Advances in neural information processing systems*, pp. 6626–6637, 2017.