Vessel Layer Separation of X-ray Angiographic Images using Deep Learning Methods

Haidong Hao

Abstract

Percutaneous coronary intervention is a minimally-invasive procedure to treat coronary artery disease. In such procedures, X-ray angiography, a real-time imaging technique, is commonly used for image guidance to identify lesion sites and navigate catheters and guide-wires within coronary arteries. Due to the physical nature of X-ray imaging, photon energy undergoes absorption when penetrating tissues, rendering a 2D projection image of a 3D scene, in which semi-transparent structures overlap with each other. The overlapping structures make robust information processing of X-ray images challenging. To tackle this issue, layer separation techniques for X-ray images were proposed to separate those structures into different image layers based on structure appearance or motion information. These techniques have been proven effective for vessel enhancement in X-ray angiograms. However, layer separation approaches still suffer either from non-robust separation or long processing time, which prevent their application in clinics.

The purposes of this work are to investigate whether vessel layer separation from X-ray angiography images is possible via deep learning methods and further to what extent vessel layer separation can be achieved with deep learning methods.

To this end, several deep learning based methods were developed and evaluated to extract the vessel layer. In particular, all the proposed methods utilize a fully convolutional network (FCN) with two different architectures (Appendix A and Chapter 2), which was trained by two different strategies: conventional losses (Appendix A and L_1 method in Chapter 2) and an adversarial loss ($AN + L_1$ method in Chapter 2).

The results of all the methods show good vessel layer separation on 42 clinical sequences. Compared to the previous state-of-the-art, the proposed methods have similar performance but runs much faster, which makes it a potential real-time clinical application. Both the L_1 method and $AN + L_1$ method in Chapter 2 achieve better background than the method proposed in Appendix A, which can remove catheter and other tubular structures well. On the other hand, the L_1 method results in better contrast and clearer backgrounds than the $AN + L_1$ method.

Acknowledgments

Studying how to apply engineering technologies to healthcare is my most interested domain, so I would like to extremely thank Prof. dr. Wiro Niessen and Dr. ir. Theo van Walsum for giving me an opportunity to do my master thesis in Biomedical Imaging Group Rotterdam, Erasmus MC.

I also would like to express my sincerest gratitude to my supervisors, Dr. ir. Theo van Walsum, Dr.ir. R.F. Remis and Hua Ma, for their guidance and patience. It is my great pleasure to work with and learn from them. I would like to thank my master program coordinator Dr.ir. R. Heusdens for helping me with my whole study process at TU Delft. In addition, many thanks to all members of both Biomedical Imaging Group Rotterdam at Erasmus MC as well as circuits and systems group at TU Delft for interesting presentations and smiles. I would like to thank my family for their emotional support and unconditional trust as well.

Lastly, I also acknowledge SURF (Collaborative organisation for ICT in Dutch education and research) for providing computing resources on SURFsara of the Cartesius system.

Haidong Hao
Rotterdam, the Netherlands
20-06-2018

Vessel Layer Separation of X-ray Angiographic Images using Deep Learning Methods

by

Haidong Hao

to obtain the degree of Master of Science
in Electrical Engineering
at the Delft University of Technology,
to be defended publicly on Thursday August 30, 2018 at 10:00 AM.

Student number:	4461703	
Project duration:	September 1, 2017 – August 30, 2018	
Thesis committee:	Dr. ir. R. F. Remis,	TU Delft, supervisor
	Dr. Anna Vilanova,	TU Delft
	Dr. Ir. Theo van Walsum,	Erasmus MC, supervisor
	Msc Hua Ma,	Erasmus MC

TU Delft

Contents

1 General Introduction

1.1. Clinical Background

According to a report of World Health Organization (WHO), ischaemic heart disease had been one of the topmost causes of death in the 15 years before 2016 [6], which is also called coronary heart disease (CHD) [13]. CHD is a disease caused by narrowing of the coronary arteries and then the narrowing reduces the blood supply to heart muscles. The artery narrowing is usually caused by atheroma, which is like fatty plaques and develops inside the arteries. As the atheroma gradually becomes larger and larger and then causes enough narrowing, people will suffer from the symptoms of angina. Figure 1.1 shows the heart location, a healthy coronary artery and narrowed coronary artery, respectively.

A treatment to widen the narrowed sections of coronary arteries is percutaneous coronary intervention (PCI), which is evolved from percutaneous transluminal coronary angioplasty (PTCA) first performed by *Andreas Grüentzig* in Zurich, on September 16, 1977 [35]. To start a PCI, the first critical step is to access the radial artery or femoral artery through a small cut in the skin by an arterial sheath. Second, a guiding catheter is inserted into an artery through the arterial sheath and then the catheter tip is pushed to the position of the coronary ostium. Next, a steerable guidewire is advanced into the coronary artery through the guide catheter and across the stenosis into the distal coronary artery. A balloon catheter is placed on the guidewire

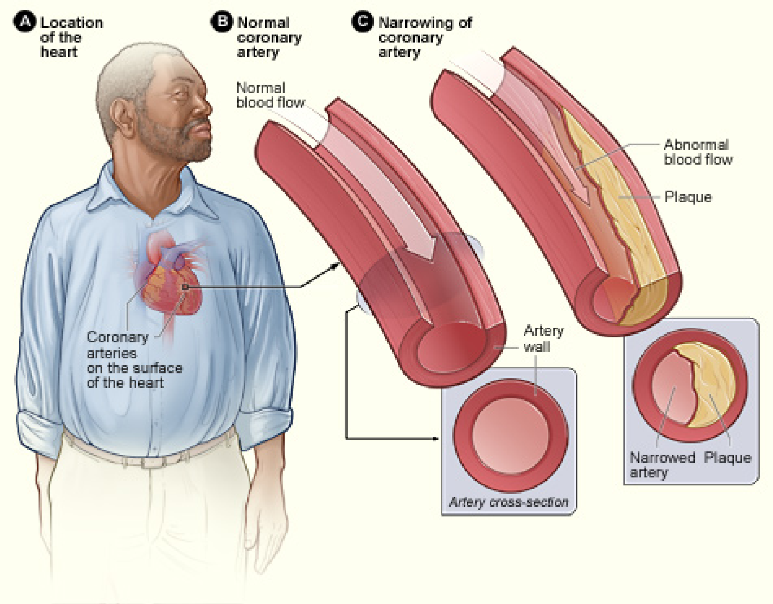

Figure 1.1: Figure A illustrates the heart location in the body. Figure B indicates a healthy coronary artery with normal blood flow; the inset image demonstrates the cross-section of the healthy coronary artery. Figure C illustrates a narrowed coronary artery with limited blood flow; the inset image is the cross-section of the narrowed artery. [1]

and advanced to the stenotic area through the guiding catheter by tracking the balloon. When the balloon is at the correct narrowed area, it will be inflated several times to expand the stenosis. After expanding the narrowed area, the balloon catheter is exchanged for the other balloon catheter mounting a compressed stent. The second balloon catheter is advanced to the stenosis using the same method as that of the first balloon catheter. After deploying the stent by inflation of the balloon, clinicians will deflate the balloon and pull the catheter out of the patient, leaving the stent in the artery to hold the artery open. A brief procedure for PCI is shown in Figure 1.2 [35].

During PCI, the guiding catheter, guidewire and both the balloon catheter have to be advanced to the proper positions inside the blood vessel, but the human body is not transparent. Therefore, an interventional X-ray system that captures X-ray angiography is used to visualize the blood vessels, localize the stenosis, and enable clinicians to navigate catheters and guidewires within the coronary artery. Figure 1.3 illustrates an overview of an interventional X-ray system.

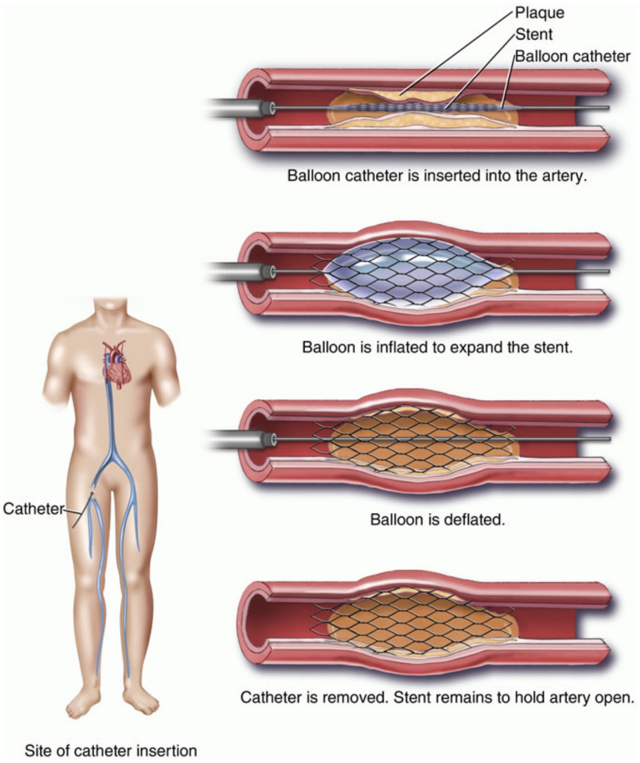
Plaque
Stent
Balloon catheter
Balloon catheter is inserted into the artery.
Balloon is inflated to expand the stent.
Catheter
Balloon is deflated.
Catheter is removed. Stent remains to hold artery open.
Site of catheter insertion

Figure 1.2: A brief procedure for percutaneous coronary intervention (PCI). [2]

Since X-ray image formation is based on photon energy absorption of various tissues along the rays, the image can be seen as a superposition of 2D projections of 3D anatomical structures, such as spine, lung and heart, which become opaque or semi-transparent structures in X-ray images. These structures normally overlap with each other and small vessels usually have low contrast, which makes accurate visualisation and quantification of the vessels as well as stenoses in X-ray angiograms (XA) challenging [22]. To handle these problems, this work aims to extract the vessel layers from XA and filter out other structures to enhance vessels.

1.2. Previous Methods

To lessen over-projection of other anatomical structures and ameliorate small vessel delineation, several approaches for vessel enhancement were proposed, such as Hessian-based vessel enhancement filters, layer separation techniques and machine learning based methods.

1.2.1. Hessian-based Vessel Enhancement Filters

Hessian-based vessel enhancement filters, such as these proposed by Lorenz et al. [41], Sato et al. [58] and Frangi et al. [22], were attempted to apply to image-guidance applications based on X-ray images (e.g. Baka et al. [10, 11], Panayiotou et al. [52], Rivest-Henault et al. [56], Wu et al. [70]). These filters calculate

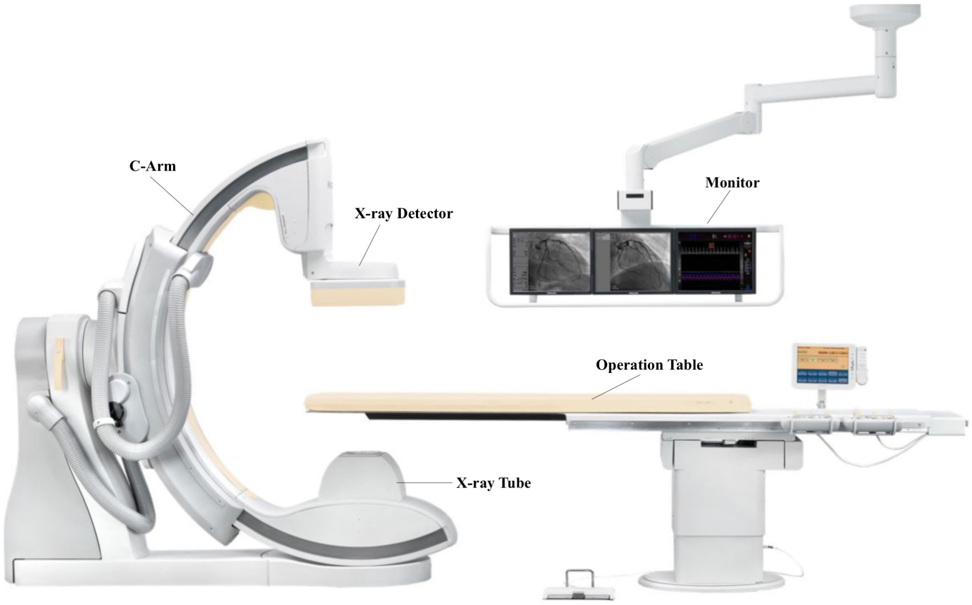

Figure 1.3: An overview of an interventional X-ray system. [3] It consists of a C-arm mounting an X-ray detector and an X-ray tube, an operation table, and a set of monitors.

second order derivatives of image intensity and apply eigenvalue decomposition to determine the likelihood of a position in images belonging to a vessel, as discussed by Olabarriaga et al. [51]. Some details about Hessian-based vessel enhancement filters are explained below.

The Taylor expansion to second order in the neighbourhood $(-s, s)$ of a point i_0 at location (x_0, y_0) is usually used to analyse local behaviors of an image, I [22],

$$I(i_0 + \Delta_{i_0}, s) \approx I(i_0, s) + \Delta_{i_0}^T \nabla_{0,s} + \Delta_{i_0}^T \mathscr{H}_{0,s} \Delta_{i_0} \tag{1.1}$$

in which, $\nabla_{0,s}$ and $\mathscr{H}_{0,s}$ are the gradient and Hessian matrix at the point i_0, respectively. The grey value of the point i_0 can be defined as $f(x_0, y_0)$, and then,

$$\mathscr{H}_{0,s} = |\frac{\partial^2 f}{\partial x_0 \partial y_0}| \tag{1.2}$$

In XA, the pixel intensity rapidly decreases from one border to the centerline of vessels and then increases to the other border. On the other hand, the pixel intensity deviates insignificantly along vessels. This can be reflected by a large positive eigenvalue (λ_1) and a small positive or negative eigenvalue (λ_2) of the Hessian matrix defined in Equation 1.2. Generally, Gaussian blurring that is convolving the image with a Gaussian function with standard deviation σ, is applied prior to calculate the Hessian matrix to reduce the influence of noise. To compensate for the decrease of derivatives caused by the Gaussian blurring, a manually defined factor σ^γ with $\gamma > 0$ was multiplied to $\frac{|\lambda_1|}{|\lambda_2|}$ and then a measure called vesselness μ was defined to verify the presence of a vessel structure at the point i_0 [41].

$$\mu = \sigma^\gamma \frac{|\lambda_1|}{|\lambda_2|} \tag{1.3}$$

However, the filters are based on the knowledge that vessels are curvilinear structure, so nearly all the other curvilinear structures, such as vertebrae, diaphragms and catheters, will be also enhanced except for vessels. Therefore, some post-processing procedures (Baka et al. [10, 11], Schneider and Sundar [59]) were employed to filter out these vessel-like structures.

1.2.2. Layer Separation Techniques

Hessian-based Vessel Enhancement Filters with post-processing methods just process a single image each time, so they can not take advantage of temporal information in XA, which can help to distinguish moving and non-moving targets. To utilize the temporal information, layer separation techniques were introduced to separate structures in X-ray images into different layers so that each layer contains structures of similar appearance or motion pattern. Existing layer separation methods for X-ray fluoroscopic sequences can be generally grouped into two classes [43]: motion-based methods (e.g. Zhang et al. [76]Zhu et al. [78]) which rely on estimation of the layered motion, and motion-free approaches (e.g. Ma et al. [42, 43], Tang et al. [63], Volpi et al. [65]) that do not require to estimate the layered motion.

Motion-based methods process each frame in XA into several layers with different motions based on assumptions that each layer only consists of structures with a similar motion type. For instance, Zhang et al. [76] assumed each frame in XA contains three motion patterns: a static background motion, a lung motion and a motion of vessels. Zhu et al. [78] only considered two layers: a coronary (vessel) layer and a background layer, using a Bayeisan framework combing dense motion estimation, uncertainty propagation and statistical fusion together to perform layer separation.

Motion-free methods separate each frame in XA into a background layer and a foreground (vessel) layer based on certain hypotheses that are used to directly model either or both of the two layers, according to the whole XA sequence. For example, Tang et al. [63] proposed a method that assumed the background and foreground are reconstructed from independent signals that are mixed together, and, therefore, the layer separation problem is equivalent to a blind source separation problem which can be solved by independent component analysis (ICA) [30].

Except for ICA, robust principal component analysis (RPCA) also can be used for source decomposition. Ma et al. [42] proposed an approach using morphological closing for breathing structures removing and then RPCA to separate XA frames into a quasi-static layer and a vessel layer. However, this method can only work in a retrospective setting, because it requires a whole sequence. Further, they extended this method to an automatic online layer separation approach by integrating online RPCA (OR-PCA) into the layer separation scheme [43].

1.2.3. Machine Learning

As discussed above, both Hessian-based Vessel Enhancement Filters and Layer Separation Techniques are based on artificially defined formal and mathematical rules, which can be named as rule-based methods [24]. However, it is difficult for human to specify perfect rules for complicated problems. Fortunately, machine learning, especially deep learning provides a data-driven strategy to solve problems. Classic machine learning methods are able to extract patterns from raw data based on hand-designed features, to solve problems, such as logistic regression proposed by Kleinbaum and Klein [37], Peduzzi et al. [54]. However, such classic machine learning methods heavily depend on the hand-designed features of the raw data, which rely on the designer's ability but not the raw data. One solution to such a problem is to utilize machine learning to discover not only the mapping from features to output but also the features themselves, automatically, which is known as representation learning. One of the representative examples of representation learning is the autoencoder, which converts the input into features by an encoder function and then convert the features into the original format by a decoder. Sometimes, it is very difficult to extract high-level, abstract features from raw data directly. Deep learning solves this problem by extracting these high-level, abstract features from other simpler features. Figure 1.4 shows a high-level schematic of the relationship of different parts of various methods described above.

So far, it has been reported that deep learning methods can achieve outstanding performances in many medical imaging tasks, including layer separation and vessel enhancement in X-ray images (e.g. Albarqouni et al. [7], Nasr-Esfahani et al. [47, 48]), by using convolutional neural networks (CNN). Apart from CNN methods, Goodfellow et al. [23] presented a new network architecture, generative adversarial networks (GAN), which follow a new strategy to train neural networks and may further boost the performances.

This thesis presents two deep learning methods for vessel layer separation, which are a fully convolutional network and an adversarial network.

1.3. Thesis Aims and Outline

In this work, deep learning methods are developed, utilized, and evaluated for vessel layer separation in clinical XA, synthetic low contrast XA, and synthetic low dose XA. These methods are compared to traditional methods and each other.

In Appendix A, we develop and evaluate a deep learning based method to extract the vessel layer. More specifically, U-Net [57], a fully convolutional network architecture, was trained to separate the vessel layer from the background. The results of our experiments show good vessel layer separation on 42 clinical sequences.

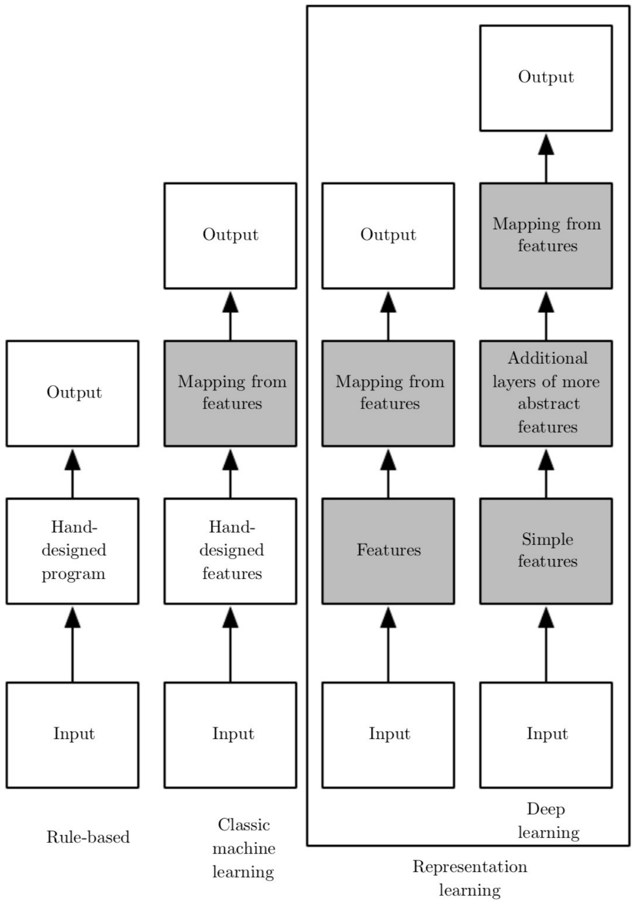

Figure 1.4: Flowcharts showing the relationship of different parts of various methods. Shaded boxes indicate components that are able to learn from data. [24]

Compared to the previous state-of-the-art, our proposed method has similar performance but runs much faster, which makes it a potential real-time clinical application.

In Chapter 2, we develop and evaluate another deep learning based layer separation method for vessel enhancement. In particular, the method utilizes a modified fully convolutional network with encoder-decoder architecture based on that explained in chapter 2, which was trained by two different strategies: a simple L_1 loss and the combination of L_1 and adversarial losses, respectively.

1.4. Contributions

The contributions of this thesis include the following:

1. proposing two (U-Net and GAN) deep learning based approaches for layer separation in XA.
2. comparing the proposed methods with the previous state-of-the-art approach [43] and each other.
3. assessing the proposed methods for low-contrast / low-dose scenarios with synthetic XA data, and show robust performance.

2

Layer Separation in X-ray Angiograms for Vessel Enhancement with Fully Convolutional Networks[1]

2.1. Introduction

2.1.1. Background

Percutaneous coronary intervention (PCI) is a minimally invasive procedure for treating patients with coronary artery disease in clinical routine. These procedures are performed under image-guidance using X-ray angiography, in which coronary arteries are visualized with X-ray radio-opaque contrast agent. Such imaging setups enable clinicians to observe coronary arteries and navigate medical instruments during interventions.

An X-ray image is a superimposition of 2D structures projected from the 3D anatomical structures. The overlapping nature of these structures in X-ray angiograms (XA) makes robust information processing challenging. For instance, Hessian-based vessel enhancement filtering (e.g. Frangi et al. [22], Sato et al. [58]), a common basic step for various image-guidance applications based on X-ray images (e.g. Baka et al. [10, 11], Panayiotou et al. [52], Rivest-Henault et al. [56], Wu et al. [70]), however, is often hampered by enhancing curvilinear structures in the background, such as diaphragm, vertebra or catheters. Though post-processing steps were used to remove spurious background structures in the vesselness images (Baka et al. [10, 11], Schneider and Sundar [59]), they are difficult to robustly scaled to large sets of data. Therefore, it is relevant to develop robust methods to automatically enhance structures of interest in XA, such as coronary arteries, while ignore other curvilinear structures in the background.

While Hessian-based vessel enhancement with post-processing steps treats one single image each time, temporal information in XA sequences, which is valuable for distinguishing moving and background structures, is not used. Layer separation was proposed for separating 2D overlapping structures in XA and putting them in different image layers by exploiting temporal information. As a result, structures with similar motion pattern or appearance are grouped together, and are ready for further analysis without interference of structures in other layers Ma et al. [43].

Traditional layer separation methods generally fall under two categories: motion-based and motion-free. Motion-based methods (e.g. Auvray et al. [9], Close et al. [18], Fischer et al. [21], Preston et al. [55], Zhang et al. [76], Zhu et al. [78]) are developed upon the basis that each XA frame is the outcome of the layered motion in the sequence, hence it is essential to accurately estimate the motion of each layer. Whereas motion-free methods (e.g. Jin et al. [33], Ma et al. [42, 43], Tang et al. [63], Volpi et al. [65]) exempt from the need of motion estimation, they directly model the layers with specific assumptions. Among these methods, only Ma et al. [43] runs online (process one frame each time without the need of future frames) and does not rely on a collection of fluoroscopic images acquired in advance, which fulfills the requirement of intra-operative use. However, this method relies on online robust principal component analysis (RPCA) algorithm (Feng et al. [20]), which is limited under noisy condition with low-dose X-ray images.

[1] Haidong Hao, Hua Ma, and Theo van Walsum. Parts of this chapter were included in a paper that was accepted by **Joint MICCAI-Workshops on Computing and Visualization for Intravascular Imaging and Computer Assisted Stenting (CVII-STENT)**.

In contrast to the traditional methods, methods based on supervised machine learning, particularly deep learning, have been reported to gain excellent performance in various medical imaging tasks Litjens et al. [39], including layer separation and vessel enhancement / segmentation in X-ray images. For example, Nasr-Esfahani et al. [47, 48] used convolutional neural networks (CNN) to extract vessels in XA based on patch-based approaches. Albarqouni et al. [7] revealed latent structures in X-ray radiographs with in-depth decomposition using a U-Net like architecture (Ronneberger et al. [57]). Hao et al. [26] used a U-Net like network to generate a saliency map where vessels are enhanced compared to the original XA.

In this scenario, layer separation or vessel enhancement is viewed as an image-to-image translation problem, in which a mapping function is learned to translate an input image (X-ray image) to an output image or a series of output images (vessels). For example, Hao et al. Hao et al. [26] used a U-net architecture to generate from the original XA to a saliency map where vessels are enhanced. Performance of image-to-image translation may be further boosted with generative adversarial networks (GANs, Goodfellow et al. [23]). GANs consist of two networks, a generator and a discriminator. The generator is trained to generate "real-like" samples from input(s), trying to fool the discriminator; while the discriminator is trained to determine as well as possible if the generated sample is from the same distribution of the real sample or not. The idea of adversarial training has been applied and achieves good performances in various medical imaging problems, such as medical image synthesis (Bayramoglu et al. [12], Chartsias et al. [16], Costa et al. [19], Hu et al. [28], Nie et al. [49], Wolterink et al. [68]), segmentation (Moeskops et al. [46], Yang et al. [73], Zhang et al. [77]), noise reduction (Wolterink et al. [69]), motion modeling (Hu et al. [29]). Nevertheless, to what extent it can be used for layer separation for vessel enhancement in X-ray images has not been explored yet.

2.1.2. Overview and Contributions

In this chapter, we investigate and evaluate a deep learning based layer separation method for vessel enhancement in XA, including trained by adversarial networks ($AN + L_1$ method) introduced in [32] and a conventional L_1 loss (L_1 method). In particular, the work focuses on transforming the XA images directly to the vessel layer where structures of interest (vessels, catheter tip, guidewire) are enhanced, and background structures (bones, diaphragm, guiding catheter) are removed. Our contributions are the following:

1. proposing a GAN-based approach ($AN + L_1$ method) for layer separation in XA.
2. comparing the proposed method with one state-of-the-art approach [26].
3. assessing the proposed methods for low-contrast / low-dose scenarios with synthetic XA data, and show robust performance.

2.2. Method

While the original GAN [23] generates new samples from random noise z, we adopt the approach introduced in [32] that trains a generator to generate a new image y from the input image x and a random noise z. Different from [32], our approach does not include the random noise z in the generator input in which the randomness is implicitly contained in the variety of the input images. Therefore, we denote the generator G in our approach as a mapping $G : x \to y$, where x is an input XA and y represents the desired output vessel layer. The method overview is illustrated in Figure 2.1.

2.2.1. Training objective

The GAN objective of our approach can be described as Equation 2.1,

$$\mathscr{L}_{GAN}(G,D) = E_{x,y\sim p_{data(x,y)}}[logD(x,y)] + E_{x\sim p_{data(x)}}[log(1-D(x,G(x)))] \tag{2.1}$$

where G is the generator, D is the discriminator, x and y denote the input XA and the reference vessel layer, respectively. Note that $\mathscr{L}_{GAN}(G,D)$ is equivalent to the binary cross-entropy loss of D for real (the first term) and fake (the second term) image pairs.

Traing the generator can be also benefited from adding an additional term for G to the GAN objective, e.g. the L_1 ([32]) or L_2 ([69]) distance, penalizing the generator output being different from the reference. We choose the L_1 distance (see Eq. 2.2) for our approach, as it preserves finer details in the images than L_2, which is advantageous to small vessel enhancement.

$$\mathscr{L}_{L_1}(G) = E_{x,y\sim p_{data}(x,y)}||y - G(x)||_1 \tag{2.2}$$

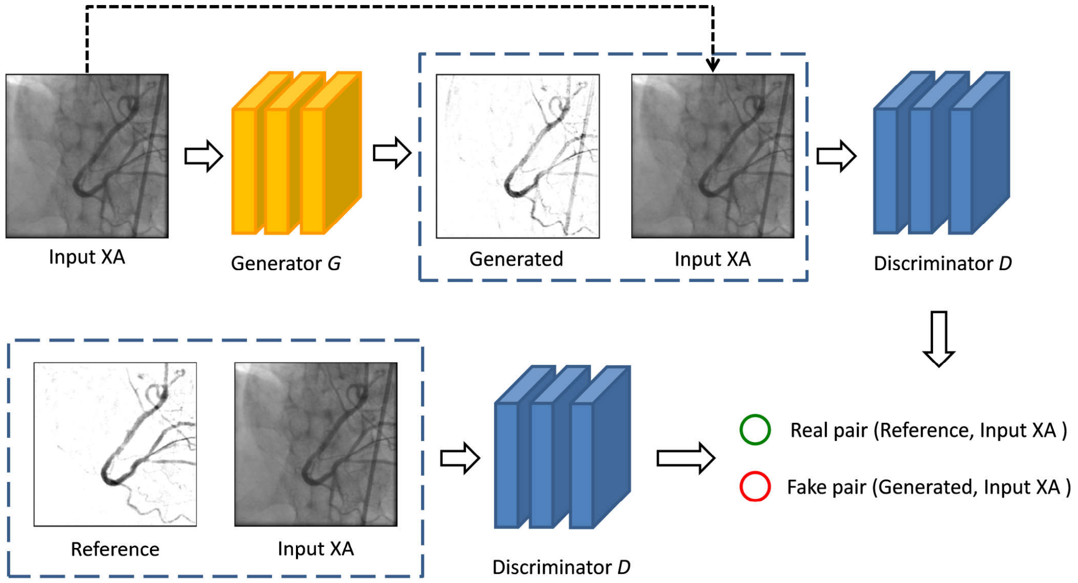

Figure 2.1: Overview of our approach. The generator G learns a pixel-to-pixel transformation that maps an input XA to a vessel layer where the vessel structure is enhanced and the background structures are removed. The Discriminator D receives the input XA and the vessel layer as an input pair. D is trained to distinguish whether the input pair is a real pair (input XA, reference vessel layer) or a fake pair (input XA, generated vessel layer). During training, D provides feedback for training G; G is trained to confuse D. Once training is done, only G is used for inference to generate vessel layer from input XA.

The total objective of our approach is expressed in Equation 2.3, where λ is a weight balancing the two terms. The training details can be referred to Algorithm 1.

$$\underset{G}{min}\underset{D}{max}\mathscr{L}_{GAN}(G,D)+\lambda\mathscr{L}_{L_1}(G) \tag{2.3}$$

2.2.2. Generator *G*

We used a U-Net-like architecture [57] for G, slightly modified from our previous work [26]. First, batch normalization [31] was applied after all convolutional layers except for the last output layer. Second, all ReLU activations were replaced by leaky ReLU with a leak slope of 0.2. Third, all max pooling layers were replaced by a stride-2 convolutional layer for spatial downsampling. The second and third point are to avoid sparse gradient. In addition, tanh activation was used as the final output of G. We also added three dropout layers in the decoding path [32] to reduce overfitting. The generator architecture can be referred to Figure 2.2.

The input x for G As XA sequences are time series of XA images, temporal information between frames is useful for distinguishing foreground and background structures in XA images. We used as the input x for G not only the current frame, but also information of a few frames before the current one, so that the output of G is conditioned on multiple frames. In particular, we used the following as different input channels of x: the current frame I_t;the difference between the current frame and its preceding frame, $d_{t-1} = I_t - I_{t-1}$; and the difference between the current frame and the first frame in the sequence, $d_1 = I_t - I_1$. The number of input channels determines the dimension of convolution kernel in the first layer of G. We did not try more previous frames, as this would cause more delay on receiving the input, which is not desirable for prospective processing of XA sequences. Example images of different input channels are shown in Figure 2.3.

The training reference y for G According to Equation 2.2, training G (also D according to Equation 2.1) requires a training reference y. To create the training reference, we used the layer separation approach in Ma et al. [42] to generate the "ground truth" vessel layer. The pixel values of the resulted vessel layer were then normalized to the range of -1 to 1 due to the last activation layer of G (see Section 2.2.2). Example images of the training reference y are shown in Figure 2.4.

Algorithm 1 Training the adversarial networks G and D

```
Require: $m$ (batch size), $n$ (number of training iterations), $\{(x^k, y^k)|k = 1 \dots N\}$ (training data), $G$ (the generator), $D$ (the discriminator)
  for i = 1 to n do
    if $i$ is odd then
      • Sample randomly $m$ image pairs $\{(x^{(1)}, y^{(1)}), ..., (x^{(m)}, y^{(m)})\}$.
      • Update the parameters of $D$ by:
          $$\max_D \frac{1}{m} \sum_{j=1}^{m} log(D(x^{(j)}, y^{(j)})   (2.4)$$
    else
      • Sample randomly $m$ XA images $\{x^{(1)}, ..., x^{(m)}\}$.
      • Update the parameters of $D$ by:
          $$\max_D \frac{1}{m} \sum_{j=1}^{m} log(1 - D(x^{(j)}, G(x^{(j)})))   (2.5)$$
    end if
    • Sample randomly $m$ image pairs $\{(x^{(1)}, y^{(1)}), ..., (x^{(m)}, y^{(m)})\}$.
    • Update the parameters of $G$ by:
          $$\min_G -\frac{1}{m} \sum_{j=1}^{m} log(D(x^{(j)}, G(x^{(j)}))) + \frac{\lambda}{m} \sum_{j=1}^{m} ||y^{(j)} - G(x^{(j)})||_1   (2.6)$$
  end for
  return $G, D$
```

2.2.3. Discriminator D

The discriminator D works as a classifier to distinguish as well as possible if its input is from the same distribution of the reference data or the generated data. The network architecture of D consists of 7 3 × 3 convolutional layers of stride-2, following by a fully-connected layer and a softmax layer. Batch normalization and leaky ReLU with a leak slope of 0.2 were used after each convolutional layer. The discriminator has several variants according to the convolutional layer number, and details of a 7-convolutional-layer discriminator architecture can be referred to Figure 2.5.

The input for D Unlike the original GAN (Goodfellow et al. [23]) in which the discriminator take only the generator output or the real sample as the input, we follow the approach in Isola et al. [32] to enforce structure correspondence between the generator input and output. In particular, the discriminator D receives as the input a pair of images, the input XA image x with the reference vessel layer image y as the real pair, or x with the generator output $G(x)$ as the fake pair (see Figure 2.1). It is also worth noting that x may be enhanced by concatenating multiple channels that contain temporal information (Section 2.2.2). The number of input channels determines the dimension of convolution kernel in the first layer of D.

2.3. Experiments

2.3.1. Data

Anonymized image data from clinical routine were acquired from Erasmus MC in Rotterdam, the Netherlands. 42 XA sequences were acquired with Siemens AXIOM-Artis biplane system from 21 patients undergoing a PCI procedure. The frame rate of all sequences is 15 frames per second (fps). All 42 sequences contain 4884 frames in total. After removing the first frame of each sequence to generate d_{t-1} and d_1, we selected 8 sequences (940 frames) as test data and the other 34 sequences were divided into five nearly equal parts (780, 786 , 772, 758, 806 frames from 5, 7, 7, 6, 9 sequences, respectively) for cross-validation.

Since the raw clinical data varies in image size, illumination, etc., two preprocessing steps were applied on the clinical data prior to processing them with the neural networks: 1) all images were resampled to the grid of

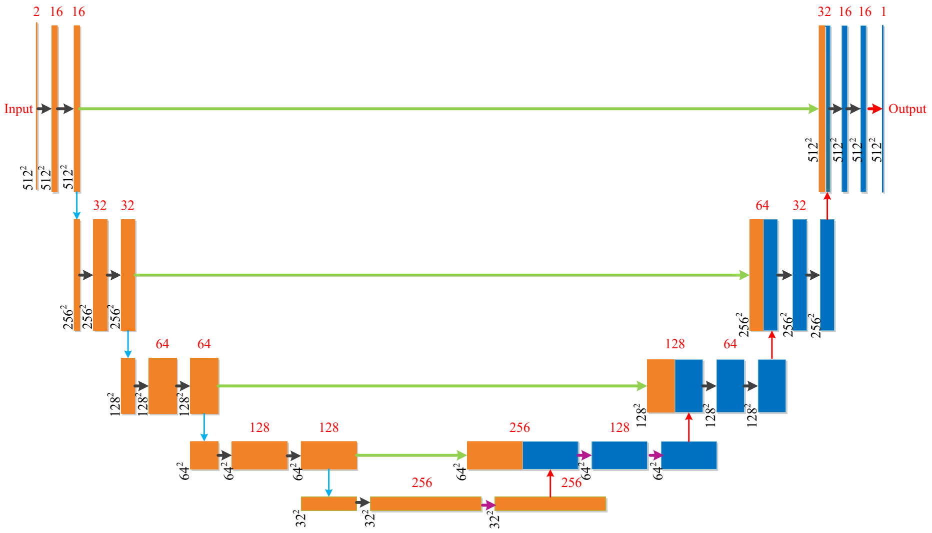

Figure 2.2: Generator architecture. The left and right sides are an encoder and a decoder, respectively. Each box denotes a feature map; the number on top of each box indicates the number of feature maps; the number at the lower left edge of each box is the size of corresponding feature maps; different color arrows denote different operations (black arrow: 3 × 3 convolution with batch normalization and a leaky ReLU activation, successively, blue arrow: 3 × 3 convolution with stride 2, green arrow: skip connection, red up arrow: upsampling operation followed by a 2 × 2 convolution without activation, purple right arrow: 3 × 3 convolution with batch normalization, dropout and a leaky ReLU activation, successively, red right arrow: 1 × 1 convolution filter with a Tanh activation); the orange boxes in the decoder represent corresponding copied feature maps from the encoding path.

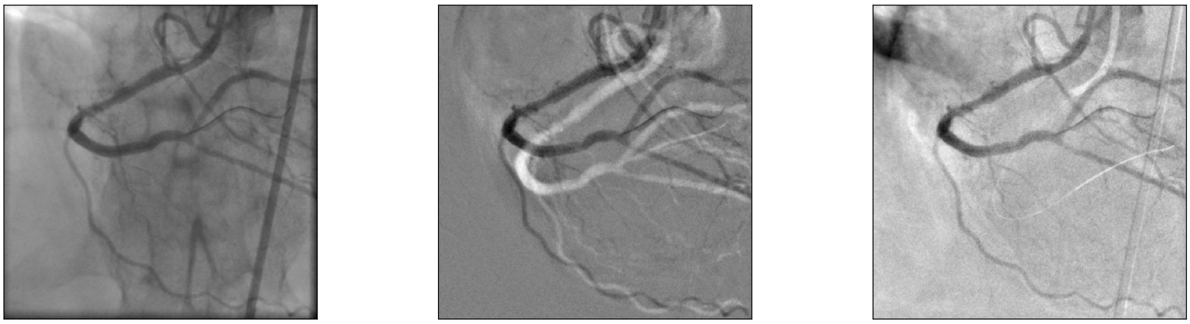

Figure 2.3: An example of each input channel from clinical angiogram. (left) the current frame (I_t); (middle) the difference between the current frame and its preceding frame ($d_{t-1} = I_t - I_{t-1}$); (right) the difference between the current frame and the first frame in the sequence ($d_1 = I_t - I_1$).

512 × 512 so that input images to the neural networks are of the same dimension while not losing much fine details from the original resolution; 2) the pixel values of all images were normalized to the same range from -1 to 1 in this work.

2.3.2. Evaluation metrics

After normalizing the range of references and predictions to from 0 and 1, we evaluate the quality of the vessel layer images using contrast-to-noise ratio (CNR, see Equation 2.7) between the foreground and background::

$$CNR = \frac{|\mu_F - \mu_B|}{\sigma_B} \tag{2.7}$$

where μ_F and μ_B denote the average intensity of the foreground and background pixels, respectively; σ_B represents the standard deviation of the background pixels.

The foregound and background areas used for computing CNR were confined by manually generated masks using the method in Ma et al. [43]. The foreground area was defined as a 1 mm wide area around manually labeled vessel centerlines, as shown in Figure 2.6 (left). This area well represents the vessel pixels, as it falls

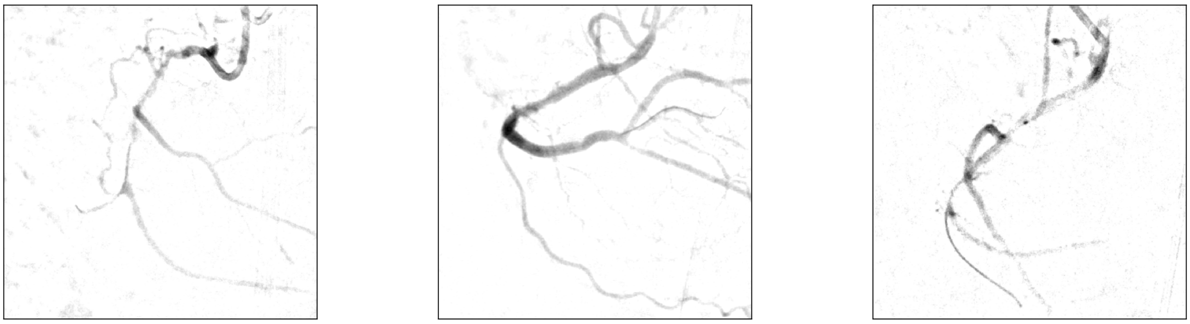

Figure 2.4: Three examples of training reference y.

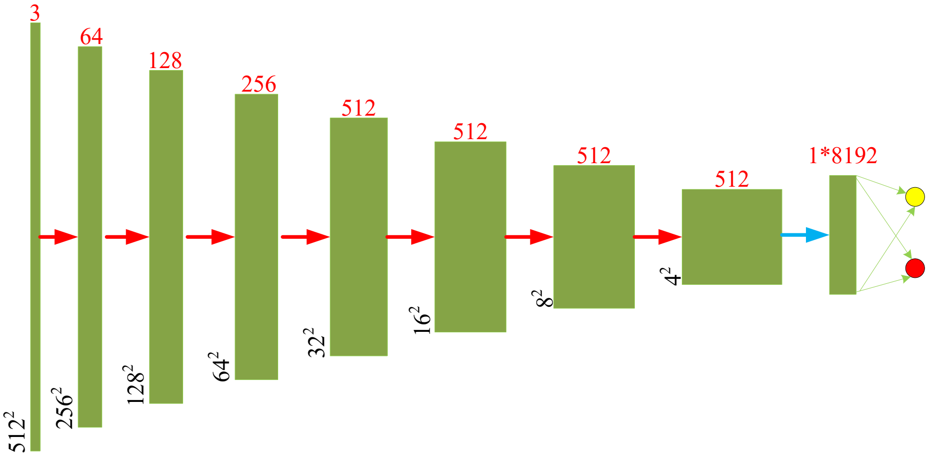

Figure 2.5: 7-convolutional-layer discriminator architecture. The discriminator is a encoder. Each green box denotes a feature map; the number on top of each box indicates the number of feature maps; the number at the lower left edge of each box is the size of corresponding feature maps; different color arrows denote different operations (red arrow: 3 × 3 convolution with batch normalization and a leaky ReLU activation, successively, blue arrow: flatten, green arrow: dense with a softmax activation; The red and yellow circle denote the probabilities belonging to real pair and fake pair, respectively.

entirely within the vessels. The background was defined with two types of masks. The first one highlights all pixels outside a 4 mm wide area around the vessel centerlines (see Figure 2.6 (middle)). This mask was used for measuring the contrast in a global scale, i.e. the effect of removing diaphragm, guiding catheters, etc. The other mask defines the "local" background, a 3 mm wide area surrounding the dark area in the global mask (the white area in Figure 2.6 (right)).

As CNR is a metric based on only the output of G, we additionally used structure similarity (SSIM) to measure the similarity between the generator output and the reference. SSIM is defined as:

$$SSIM = \frac{(2\mu_r\mu_p + C_1)(2\sigma_{rp} + C_2)}{(\mu_r^2 + \mu_p^2 + C_1)(\sigma_r^2 + \sigma_p^2 + C_2)} \tag{2.8}$$

where μ_r, μ_p, σ_r and σ_p denote the mean and standard deviation of the pixel intensities in the reference and predicted images, respectively; σ_{rp} denotes the covariance of the pixel values between the references and the predictions. C_1 and C_2 are small constants to avoid zero-division in Equation 2.8. We followed the settings in Wang et al. [66], defining $C_{1,2} = (K_{1,2}L)^2$, where $K_1 = 10^{-8}, K_2 = 3 \times 10^{-8}$ and $L = 1$, the dynamic range of the pixel values in the post-processed (normalisation of 0 to 1) reference and prediction images.

As discussed in [66], Equation 2.8 can achieve the unique maximum value, if and only if the pixel value of each pixel in the prediction is equal to the pixel value of the pixel in the same position of the reference, which means $\mu_r = \mu_p$ and $\sigma_r^2 = \sigma_p^2 = \sigma_{rp}^2$. In addition, it is easy to prove that Equation 2.8 is an even function. Therefore, SSIM will decrease as $\Delta\mu = |\mu_r - \mu_p|$, and/or $\Delta\sigma^2 = |\sigma_r^2 - \sigma_p^2|$, and/or $\Delta\sigma_{rp}^2 = |\sigma_r^2 - \sigma_{rp}^2|$ increase.

Similar to CNR, we also computed SSIM in both local and global scale. The global SSIM was computed over the entire image, whereas the local SSIM was computed within the vessel area defined by the white area in the

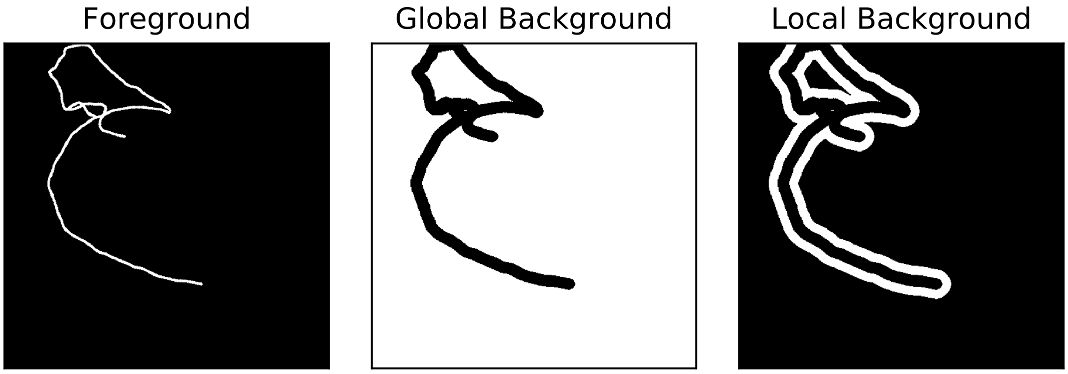

Figure 2.6: An example of foreground and background proposed by Ma et al. [43]. (left) foreground (white area); (middle) global background (white area); (right) local background (white area).

reverse of the global mask for CNR (see Figure 2.6 (middle)). For each XA sequence, we randomly selected 8-15 frames with contrast agent for contrast evaluation. The number of selected frames depends on the sequence length. In total, 444 frames were selected from 42 sequences.

2.3.3. Implementation

All the networks were trained and evaluated on SURFsara of the Cartesius system with an NVIDIA Tesla K40m GPU using Keras with Tensorflow as the backend. The parameters of all the networks were trained using an ADAM optimizer [36].

2.3.4. Experiment 1: Evaluation on clinical XA

We compared the performance of training the generator with L_1 only (Equation 2.2) and the combination of L_1 and adversarial loss (Equation 2.3). In addition, we also evaluated the influences of input channels, 1-Channel (1Ch, (I_t)), 2-Channel (2Ch, (I_t, d_1)) and 3-Channel (3Ch, (I_t, d_{t-1}, d_1)), respectively.

After tuning both the L_1 method and $AN + L_1$ method, they were compared using average CNR and SSIM of 42 frames from 4 sequence as the evaluation metrics, and with the method presented by Hao et al. [26].

2.3.5. Experiment 2: Evaluation on synthetic low-contrast XA

According to [43], layer separation has the potential to enhance vessels in X-ray images with low vessel contrast, which may be caused by obese patients or reduction of contrast agent concentration for contrast agent allergic patients. To this end, in addition to the normal clinical XA, we also evaluated our proposed method on low-contrast XA synthesized from the clinical XA, with the same reference images as those in Experiment 1. Similar to Experiment 1, we examined the influences of input channels as well and compared to the method presented by Hao et al. [26].

The synthetic images simulate a 50% lower contrast concentration and were constructed using an offline RPCA approach [43].

2.3.6. Experiment 3: Evaluation on synthetic low dose XA

Percutaneous coronary interventions may increase radiation exposure risks and then cause radiation-induced injuries to both patients and interventional cardiologists, such as skin burns and even cancer [45]. Therefore, there is a need to limit radiation exposure by low dose X-ray angiography, which may degrade the image quality [38]. To assess whether our proposed method can be utilized to decrease X-ray dosage, we evaluate the performance of our methods on synthetic low-dose XA. The reference images are the same as those in Experiment 1 and the 1Ch, 2Ch and 3Ch inputs were experimented with as well.

The synthetic low-dose XA was generated by adding Gaussian noise to the clinical XA. The mean of the Gaussian noise is 0 and the variance is dependent on the clinical XA which is calculated by Equation (2.9):

$$\sigma_n^2 = \frac{\sigma_s^2}{10^{\frac{SNR}{10}}} \tag{2.9}$$

in which, σ_n^2 and σ_s^2 are the variance of Gaussian noise and the clinical XA, respectively; SNR denotes signal-to-noise ratio and we applied 5 in this work to synthesize $SNR = 5$ low-dose dataset.

2.4. Results

2.4.1. Experiment 1: Evaluation on clinical XA

The optimal hyper-parameters obtained from cross-validation for both $AN + L_1$ and L_1 methods are shown in Table 2.1 and Table 2.2, respectively. Average CNR and SSIM of both our proposed methods and the state-of-the-art method based on the test data of clinical XA are shown in Figure 2.7. Figure 2.8 illustrates two prediction examples of these methods.

Table 2.1: The optimal hyper parameters of $AN + L_1$ method based on Clinical XA. 5-convolutional discriminator is an alternative architecture of D; Filter No. indicates the filter number in the first convolutional layer of G.

Input	**Learning rate (G)**	**Learning rate (D**	**Epoch No.**	**D Architecture**	**Filter No. (f)**	λ
2Ch	5×10^{-4}	5×10^{-4}	50	5-convolutional	16	10

Table 2.2: The optimal hyper parameters of L_1 method based on Clinical XA. Filter No. indicates the filter number in the first convolutional layer of G.

Input	**Learning rate (G)**	**Epoch No.**	**Filter No. (f)**
2Ch	5×10^{-4}	50	16

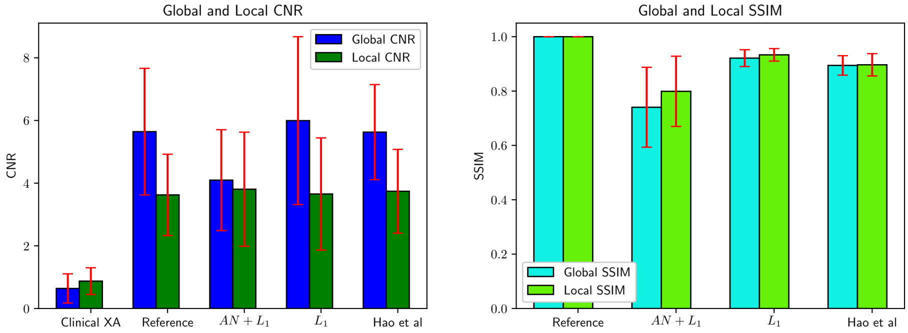

Figure 2.7: Average CNR and SSIM of various methods based on the test data of clinical XA.

As illustrated in Figure 2.7, all the three methods achieve nearly the same local CNR that is also similar to the reference. Refer to the prediction examples shown in Figure 2.8, the vessel area of the examples of $AN + L_1$ method is the brightest among all the methods as well as the reference; in terms of the background, both $AN + L_1$ and L_1 methods obtain clearer background than the other method that did not remove the catheter and some tubular structures well.

We used a two-sided Wilcoxon signed-rank test to assess whether the results are statistically significantly different. As shown in Table 2.3, $AN + L_1$ method is statistically different from L_1 method; $AN + L_1$ method and the method proposed in [26] also are statistically different except for local CNR; Differences between L_1 method and the method proposed in [26] are only statistically significant for SSIM.

2.4.2. Experiment 2: Evaluation on synthetic low-contrast XA

The optimal hyper-parameters for both $AN + L_1$ and L_1 methods based on low-contrast XA are shown in Table 2.4 and Table 2.5, respectively. Average CNR and SSIM of our proposed methods and the method

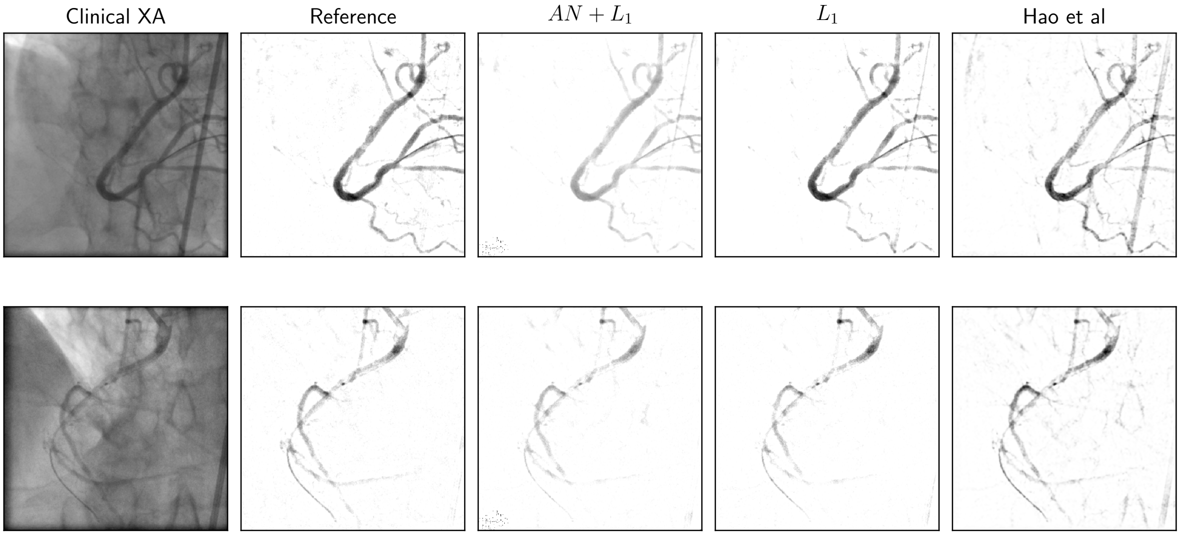

Figure 2.8: Two prediction examples of various methods base on the test data of Clinical XA.

Table 2.3: p-values among various methods in terms of average CNR and SSIM based on clinical XA.

Method 1	Method 2	Local CNR	Global CNR	Local SSIM	Global SSIM
$AN+L_1$	L_1	0.031	<0.001	<0.001	<0.001
$AN+L_1$	Hao et al. [26]	0.910	<0.001	<0.001	<0.001
L_1	Hao et al. [26]	0.181	0.424	<0.001	<0.001

presented by Hao et al. [26] based on the test data of low contrast XA are shown in Figure 2.9. Figure 2.10 illustrates two prediction examples of these methods.

Table 2.4: The optimal hyper parameters of $AN+L_1$ method based on low-contrast XA. 6-convolutional discriminator is an alternative architecture of D; Filter No. indicates the filter number in the first convolutional layer of G.

Input	Learning rate (G)	Learning rate (D	Epoch No.	D Architecture	Filter No. (f)	λ
2Ch	5×10^{-4}	5×10^{-4}	50	6-convolutional	16	10

As illustrated in Figure 2.9, all the three methods achieve nearly the same local CNR that is also similar to the reference. Refer to the prediction examples shown in Figure 2.10, the vessel area of the examples of $AN+L_1$ method is the brightest among all the methods as well as the reference; in terms of the background, both $AN+L_1$ and L_1 methods obtain clearer background than the method presented by Hao et al. [26] which did not remove the catheter and some curvilinear structures well.

We utilized two-sided Wilcoxon signed-rank test to assess whether the performances of all the methods are statistically different from each other based on average CNR and SSIM as well. The results are shown in Table 2.6. Both $AN+L_1$ versus L_1 and $AN+L_1$ versus the method proposed in [26] are statistically different except for local CNR; There are statistical differences between L_1 method and the method proposed in [26] only for SSIM.

2.4.3. Experiment 3: Evaluation on synthetic low dose XA

The optimal hyper-parameter combination of both $AN+L_1$ and L_1 method are shown in Table 2.7 and Table 2.8, respectively. Figure 2.11 illustrates average CNR and SSIM of both our methods based on the test

Table 2.5: The optimal hyper parameters of L_1 method based on low-contrast XA. Filter No. indicates the filter number in the first convolutional layer of G.

Input	Learning rate (G)	Epoch No.	Filter No. (f)
2Ch	5×10^{-4}	50	16

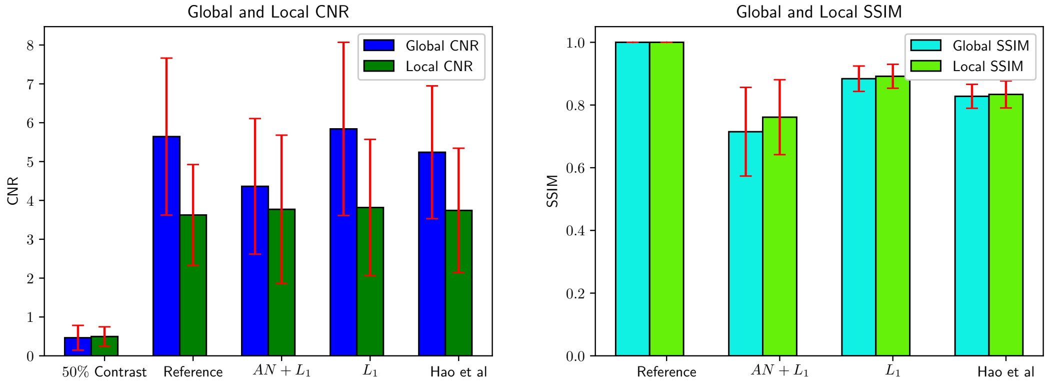

Figure 2.9: Average CNR and SSIM of various methods based on low-contrast XA

data of low dose XA. Two prediction examples of these methods are shown in Figure 2.12.

Both the two methods achieve similar local CNR to the reference and the $AN+L_1$ method obtains higher global CNR and SSIM, which can be seen in Figure 2.11. As shown in Figure 2.12, $AN+L_1$ method produced less dark vessel area, while both methods generated similar background to the reference. The two-sided Wilcoxon signed-rank test results shown in Table 2.9 indicates that the two methods are statistically different except for local CNR.

2.5. Discussion

2.5.1. Hyper-parameter tuning

As shown in Table 2.2, 2.5, 2.8, the hyper-parameters of the L_1 method based on all the three datasets are the same, while the hyper-parameters of the $AN+L_1$ method based on all the datasets are the same except for the discriminator architecture, which can be referred in Table 2.1, 2.4, 2.7. In addition, λ in Equation 2.3 is a very important hyper-parameter and we tested four optional values (10, 1, 0.1, and 0). As the value of λ decreases, the convergence speed of generator becomes slower and slower. Particularly, it is difficult to converge, when λ is equal to 0. This may indicate gradient vanishing for the gradient flow from the discriminator, because the discriminator is too deep. Another central hyper-parameter is the input, which delimits what temporal information can be utilized. As shown in Figure 2.3, stationary structures, such as catheter, were removed and the vessel layer was enhanced in d_1 (right), while a brighter vessel area was introduced maybe as noises in d_{t-1} (middle), so d_1 may help improve the performance.

2.5.2. Comparison between $AN+L_1$ method and L_1 method

As shown in Figure 2.7, 2.9, 2.11, for the three datasets, both $AN+L_1$ and L_1 methods achieve similar local CNR to the reference, which indicates that our methods can obtain robust performance. However, the vessels in the prediction examples shown in Figure 2.8, 2.10, 2.12 as well as the other 40 frames for calculating the corresponding metrics are less dark, respectively. Why this happened? Because the less darkness of the vessels can decrease both the denominator and numerator of Equation 2.7 simultaneously, which can keep the local CNR changeless. The only difference between the two methods is the loss function (Equation 2.2 versus Equation 2.3). $AN+L_1$ method updates the network parameters of G from two parts of losses (L_1 loss and adversarial loss) parallelly, in which the L_1 loss makes the output of G similar to the reference pixel-wise, but the adversarial loss forces the output of G similar to the reference globally. In addition, two optimizers were utilized to update the network parameters of G in $AN+L_1$ method, which can be regarded as adjusting the

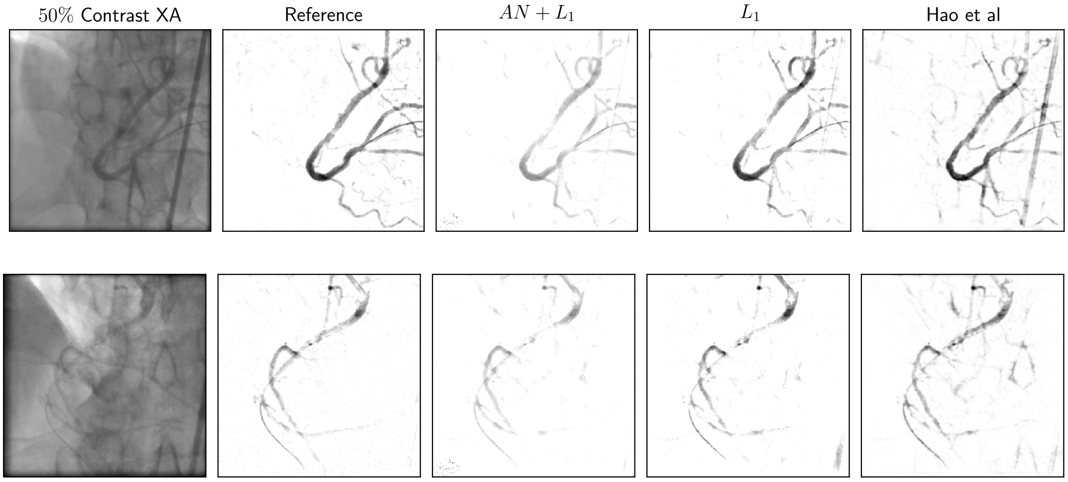

Figure 2.10: Two prediction examples of various methods base on low-contrast XA.

Table 2.6: p-values among various methods in terms of average CNR and SSIM based on low contrast XA.

Method 1	Method 2	Local CNR	Global CNR	Local SSIM	Global SSIM
$AN + L_1$	L_1	0.764	<0.001	<0.001	<0.001
$AN + L_1$	Hao et al. [26]	0.617	0.001	<0.001	<0.001
L_1	Hao et al. [26]	0.950	0.084	<0.001	<0.001

network parameters already optimized with L_1 loss by optimizing the adversarial loss, so the output of $AN + L_1$ method may be slightly different from that of L_1 method. The less darkness of the vessels also influences SSIM, which make the SSIM of $AN + L_1$ method lower than those of L_1 method.

Refer to the backgrounds in Figure 2.8, 2.10, 2.12 and the other 40 frames for metrics calculation, respectively, both the two methods achieve clear background similar to the reference base on all the datasets, which may thank for the modification of the generator architecture.

In summary, the performance of $AN + L_1$ method is slightly worse than that of L_1 method based on the test data of the three datasets. This may indicate the $AN + L_1$ method is good at semi-supervised as well as unsupervised learning but not supervised learning.

2.5.3. Comparison with the state-of-the-art method

For both clinical XA and low-contrast XA, as illustrated in Figure 2.8 and 2.10 and also the other 40 frames, respctively, the vessel area of the method proposed by Hao et al. [26] look nearly the same as L_1 method and the reference, which is also reflected by the similar values for CNR. In addition, the catheter and other tubular structures can not be completely removed in the global background of the state-of-the-art method, which mainly increases the global σ_B, and then decrease the global CNR. White area occupies the majority of the global background in comparison with the catheter and other tubular structures, the global μ_B and then $|\mu_F - \mu_B|$ are influenced minimally.

Refer to Figure 2.8 and 2.10, the vessel area of both the L_1 method and the method proposed by Hao et al. [26] are nearly the same except for some small vessels and cross points between the vessel and the catheter, resulting in slightly lower local SSIM. Similarly, because of the presence of the catheter and other tubular structures, the global SSIM is also slightly smaller.

In terms of the processing speed, both L_1 method and $AN + L_1$ method achieve a rate of about 18 fps using a modern GPU, which is faster than the common image acquisition rate in clinics (15 fps). This result indicates the potential for a real-time clinical application. This is a major advantage over previous methods that are based on offline and online RPCA: those methods, though fast, are not sufficiently fast for real-time use.

Table 2.7: The optimal hyper parameters of $AN + L_1$ method based on low-dose XA. 7-convolutional discriminator is an alternative architecture of D; Filter No. indicates the filter number in the first convolutional layer of G.

Input	**Learning rate (G)**	**Learning rate (D**	**Epoch No.**	**D Architecture**	**Filter No. (f)**	λ
2Ch	5×10^{-4}	5×10^{-4}	50	7-convolutional	16	10

Table 2.8: The optimal hyper parameters of L_1 method based on low-dose XA. Filter No. indicates the filter number in the first convolutional layer of G.

Input	**Learning rate (G)**	**Epoch No.**	**Filter No. (f)**
2Ch	5×10^{-4}	50	16

2.6. Conclusion

In conclusion, we proposed deep learning based approaches for layer separation in XA. Our experiments demonstrated that the U-net like arcihtecture trained with L_1 loss performs similar to prevoius approaches, and we also showed that an additional discriminator network does not bring added value for this application. The methods can run in real-time, and thus have potential for clinical applications in interventions.

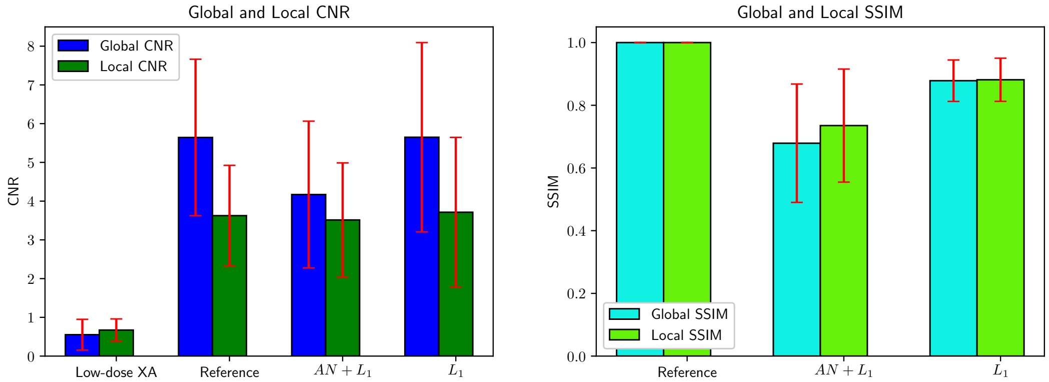

Figure 2.11: Average test CNR and SSIM of various methods based on low-dose XA.

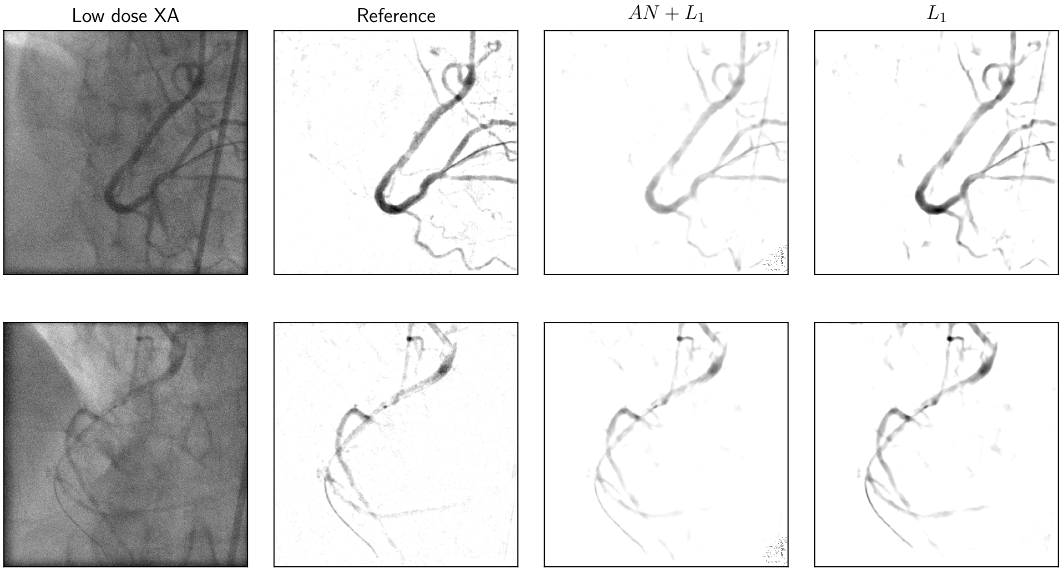

Figure 2.12: Two prediction examples of the $AN + L_1$ and L_1 methods base on low-dose XA.

Table 2.9: p-values bwtween the L_1 method and the $AN + L_1$ method in terms of average CNR and SSIM based on low dose XA.

Method 1	**Method 2**	**Local CNR**	**Global CNR**	**Local SSIM**	**Global SSIM**
$AN + L_1$	L_1	0.294	<0.001	<0.001	<0.001

3

General Discussion and Future Perspectives

The aim of this thesis is to study to what extent deep learning techniques can aid automated vessel layer separation in X-ray angiograms with different characteristics (clinical XA, synthetic low-contrast and low-dose XA), which can be formulated as an image-to-image translation problem in deep learning. It has been reported that a fully convolutional network with an encoder and a decoder has achieved excellent performance in solving image-to-image translation problems.

The deep learning based method studied in this thesis can be divided into two categories: fully convolutional network with a conventional loss (Appendix A and L_1 method in Chapter 2) and fully convolutional network with a combination of a conventional loss and an adversarial loss ($AN + L_1$ method in Chapter 2).

3.1. General Discussion

3.1.1. Compare Rule based Methods with Machine Learning based Methods

As introduced in Chapter 1, both rule based methods (Hessian-based vessel enhancement filters and layer separation techniques) and machine learning, especially deep learning based methods proposed in Chapter 2 and Appendix A can be used to draw inference from XA. Both approaches have their own strengths and weaknesses.

Rule based Methods Rule based methods utilize a series of IF-THEN statements to guide a computer to solve problems, which is built on mainly two components: a set of knowledge about the data, and a set of rules for how to utilize the knowledge [4]. As we all know, vessels in XA are dark curvilinear structures, which is the knowledge about vessels. In addition, curvilinear structures can be reflected by a specific relationship of the two eigenvalues of the Hessian matrix, which can be regarded as the rule. Therefore, Hessian-based vessel enhancement filters determines whether a pixel in XA belongs to a vessel or not based on if the value of vesselness μ shown in Equation 1.3 is larger or smaller than a threshold.

One key strength of rule based methods is that formulating and performing rules is quite easy by translating the experience and knowledge of experts [5]. On the other hand, there are special cases and/or exceptions that need to be considered, such as the presence of vertebrae, diaphragms and catheters that also have curvilinear structures. As the number of special cases and/or exceptions increases over time, rule based methods may become unwieldy. If the rules, special cases, and exceptions have not been formulated rightly, false positives and negatives will increase. However, it should be very difficult to grasp all the rules, to measure how many special cases and exceptions there are [4].

Another challenge confronted by rule based methods is that it is difficult to update the rules, when the data and scenarios change faster and faster, which will make the rules further adrift from the realities of the input data and then lead to confusing results [4].

Machine Learning based Methods Machine learning based methods is an alternative way to solve problems, which may overcome some issues caused by rule based methods. Machine learning based methods typically utilize the outcomes created by experts, rather than try to completely emulate the decision process of a best practice or an expert [4]. For example, the proposed methods in Appendix A and Chapter 2 only need some paired XA images and the corresponding reference images to train the neural network and then for inference of new XA data. It may make machine learning based methods more flexible and less susceptible to some problems faced by rule based methods by only focusing on the outcomes rather than the whole decision making process.

Machine learning based methods is probabilistic and based on statistical models (networks), different from rule based methods, which is based on deterministic rules. The general idea of machine learning based methods, especially supervised learning, is to say "given what we have known about historic outcomes, what can we conclude about future outcomes from future input" [4]. The model (network) of machine learning based methods can be regarded as a function that is used to describe the outputs of the methods based on a combination of input and the model (network) parameters. In addition, the model parameters is identified by a large amounts of historic outcomes and updated by a training process.

Furthermore, machine learning based methods is often regarded as a "black box", in which both input and output are linked to realities, but the internal processes are challenging to describe and explain. The disconnection between the realities and the internal processes may help avoid complex rules formulation and execution [4].

3.1.2. How to Build a Proper Network for a Problem?

As shown in Figure 2.2, A.1, and A.6, all the proposed neural networks for layer separation in this thesis consist of some or all of the following fundamental components (layers or operations): convolution, batch normalization, activation functions (ReLU or leaky ReLU), downsampling (pooling or convolution with stride 2), upsampling, dropout, and skip connection, although they actually have different architectures with distinct performances. Therefore, how to construct a proper network for layer separation is an essential question to answer.

Convolution Operation Machine learning has been benefited from three important ideas of convolution, which are sparse interactions, parameter sharing and equivariant representations [25].

In traditional neural networks, matrix multiplication is used to describe the interaction between each input unit and the corresponding output unit, which means that every output unit interacts with every input unit. However, convolution operation utilizes a kernel smaller than the input (e.g. 3 × 3 in this thesis) to accomplish sparse interactions. For example, when processing an XA frame (512 × 512 pixels), small and meaningful features can be detected by a kernel that only interacts with 3 × 3 pixels. Consequently, fewer parameters need to be stored and thus reduces memory requirements as well as improves the statistical efficiency of the network [25].

Parameters sharing means utilisations of the same parameter for more than one function in a network. In convolution operation, each component of the kernel is applied to every position of the input (e.g. an XA frame), perhaps except some boundary pixels, while each element of a weight matrix is exactly utilized once to compute the output in traditional neural networks. Parameters sharing only learns one set of parameters (for an XA frame or a batch of XA frames) rather than a separate set of parameters for every pixel. Thus memory requirements can be further reduced and statistical efficiencies also could be improved [25].

If the input changes, and then the output changes in the same way, we can say a function is equivariant. For example, a function $f(x)$ is equivariant to a function $g(y)$ if $g(f(x)) = f(g(x))$. Particularly with images, convolution produces a 2-D map of features emerging in the input. If objects within the input shift, their representations will shift the same amount in the output, which is called equivariant representations.

Activation Function There are many activation functions used in neural networks, such as Sigmoid, tanh, ReLU, Maxout, ELU, etc. However, a ReLU or its generalizations (Leaky ReLU [44], parametric ReLU or PReLU [27]) are better for convolutional network. ReLUs are very easy to optimise due to their similarity to linear functions. The only difference between linear functions and ReLUs is that ReLUs produce zero when the input is negative, which makes derivatives through ReLUs large and consistent (equal to 1) when they are active. Thus the gradient direction is useful for learning.

Downsampling Both pooling functions (max pooling, the average of a rectangular neighbourhood, a weighted average based on the distance from the central pixel, the L2 norm of a rectangular neighborhood) and convolution with stride 2 can be used to downsample outputs of a layer, which replace the output of a layer at a location with a summary statistic of the neighbour outputs and thus make the representation preserve features at that location [25], while remove irrelevant details. Which types of pooling can be used in a network may depend on several factors, such as the relation between the sample cardinality in a spatial pool and the resolution at which low-level features have been extracted [14].

Upsampling Upsampling is a technique which is used to upsample images or other signals to a higher resolution, such as resampling and interpolation, unpooling [74], transpose convolution [75].

Batch Normalization The parameters of a neural network layer are updated by backpropagation based on an assumption that the other layers do not change. However, all the layers are updated simultaneously in practice, which may lead to unexpected results for very deep networks. Because they consist of many functions that are composed together, which are changed simultaneously when the parameters of a certain layer is being updated. Batch normalization introduces a great technique to reparameterize any deep network, which can significantly reduce the above problem [25].

Dropout Dropout [61] is a method to regularize a broad family of models and then prevent overfitting. The primary idea of dropout is randomly dropping neurons from a layer, which results in a situation where many different models learn the relation between the input and the target. This has the effect of taking ensembles of many models, which can be regarded as training a network with stochastic behaviour and making predictions by averaging multiple stochastic decisions [25].

To summarize, because the input are XA images that has known topological structure, a fully convolutional network is a better option. Then ReLUs or their generalizations may be good activation functions in cooperation with convolution operations. In addition, an encoder is needed to extract features relevant to vessels and a decoder is successively connected to the encoder to convert the vessel features into images only containing vessel layer. Therefore, downsampling and upsampling techniques are utilized in the encoder and decoder, respectively. Batch normalization can affect optimization performance dramatically, however, it is not essential to apply batch normalization to every network. For example, the network shown in Figure A.1 (without batch normalization) can achieve similar results to the network shown in Figure 2.2 (with batch normalization). Dropout is a good regularizer to reduce generalization error.

On the other hand, for a machine learning method, especially a deep learning method, it is usually too difficult to conclude whether its poor or good performance is intrinsic to which component. Firstly, these methods can be modeled as a function to transform inputs into desired outputs, which actually can be formulated with various operation combinations; secondly, these methods have multiple components adaptive with each other, if one component is broken, the others may adapt and still achieve similar performance [25].

3.1.3. Compare Training with a Conventional Loss and an Adversarial Loss

Loss functions are an essential part of deep learning methods, which measure how well a deep neural network models a dataset and are used to find the best parameters of the deep neural network in the process of optimization. There are various conventional loss functions, such as mean absolute error (L_1), mean squared error (L_2), categorical cross-entropy, binary cross-entropy and so on. Given a particular model and dataset, each loss function may achieve different performances based on its own intrinsic properties.

L_1 Loss As shown in Equation 2.2, L_1 loss is a quite simple loss function. However, it has advantages, firstly, it definitely gives a reasonable description of the differences between predictions and references of a model; secondly, it is very effectively to optimize. L_1 loss sometimes is used for a regression problem and sometimes as a regularization term (or regularizer) added to a loss function.

Binary Cross-entropy As discussed in **Appendix A**, loss functions were used to quantify the differences between the prediction distribution $p(\hat{y})$ and the reference distribution $p(y)$. Kullback-Leibler divergence (KLD) can be used to measure the differences[25], which is shown in Equation 3.1.

$$\begin{aligned} D_{KL}(p(\mathbf{y})||p(\hat{\mathbf{y}})) &= E_{x\sim p(\mathbf{y})}([\log \frac{p(\mathbf{y})}{p(\hat{\mathbf{y}})}]) \\ &= E_{x\sim p(\mathbf{y})}[\log p(\mathbf{y}) - \log p(\hat{\mathbf{y}})] \\ &= E_{x\sim p(\mathbf{y})}[\log p(\mathbf{y})] - E_{x\sim p(\mathbf{y})}[\log p(\hat{\mathbf{y}})] \end{aligned} \tag{3.1}$$

in which, $E_{x\sim p(\mathbf{y})}[\log p(\mathbf{y})] = -H(p(\mathbf{y}))$ and $-E_{x\sim p(\mathbf{y})}[\log p(\hat{\mathbf{y}})] = H(p(\mathbf{y}), p(\hat{\mathbf{y}}))$. Since $H(p(\mathbf{y}))$ is constant, minimizing the KL divergence is equivalent to minimizing the cross-entropy $H(p(\mathbf{y}), p(\hat{\mathbf{y}}))$[25].

As shown in Figure A.2, especially in the second row, it can be assumed that any image consists of two semantic labels, which are the blood vessel and the background. The normalized pixel value of each pixel in both original angiogram and the corresponding reference image can be regarded as the probability of that pixel belonging to the background and the probability of that pixel belonging to the blood vessel is one minus the normalized pixel value[67]. Therefore, binary cross-entropy can be used as a loss function.

Adversarial Loss As proposed in [23], training with an adversarial loss is equivalent to minimize the Jensen–Shannon divergence (JSD) between the prediction distribution $p(\hat{y})$ and the reference distribution $p(y)$, theoretically, which can be referred to Equation 3.2.

$$\underset{G}{min}\underset{D}{max}\mathscr{L}_{GAN}(G,D) = min - log(4) + 2 \times JSD(p(y) \| p(\hat{y})) \tag{3.2}$$

To summarize, firstly, with respect to the prediction, both L_1 and binary cross-entropy are convex, however, an adversarial loss can not be guaranteed to be convex, so the adversarial loss is more difficult to reach the global optima than the other two losses. Secondly, the binary cross-entropy loss has connections to KLD, but the adversarial loss connects to JSD. Minimizing JSD yields a distribution that fits the main mode of a dataset well, but which ignores other parts of the dataset. On the other hand, minimizing KLD prevents assigning excessively small probabilities to any data points but assigns many probability masses to non-data regions [64], which may cause the prediction less accurate than that of minimizing JSD.

3.2. Future perspectives

The proposed deep leaning based layer separation methods may help clinicians improve the routine procedure of percutaneous coronary intervention, such as reducing contrast agent concentration or X-ray dosage, by doing some additional researches, which may be the following:

1. Because the dataset used in this thesis is only 4,884 frames from 42 sequences, which may not completely represent the distribution of X-ray angiograms, increasing the amount of data may improve performance.

2. The reference images used in this thesis are generated with an offline RPCA method, which makes the reference image noisy and inaccurate, both more accurate manual annotation and noisy label learning techniques [62] [8] [34] may be advantageous to vessel layer separation.

3. Because X-ray angiograms are temporal sequence, recurrent neural networks [17] [72] [15] may boost the performance by utilizing the temporal information.

A

Vessel Layer Separation in X-ray Angiograms with Fully Convolutional Network[1]

A.1. Introduction

A.1.1. Motivation

Percutaneous coronary intervention (PCI) is a minimally-invasive procedure to treat coronary arteries disease. In such procedures, a catheter with a pre-mounted stent is introduced to the lesion site through the femoral or radial artery; during such procedures, X-ray angiography is used to visualize the blood vessels, and enables clinicians to navigate catheters and guidewires within the coronary artery. Since X-ray image formation is based on photon energy absorption of various tissues along the rays, the image can be seen as a superposition of 2D projections of 3D anatomical structures, such as spine, lung and heart, which become opaque or semi-transparent structures in X-ray images. These structures normally overlap with each other, which makes robust information processing in X-ray angiograms difficult.

A.1.2. Related works

Layer separation techniques were introduced to separate structures in X-ray images into different layers so that each layer contains structures of similar appearance or motion pattern. Existing layer separation methods for X-ray fluoroscopic sequences can be generally grouped into two classes [43]: motion-based methods[76][78] which rely on estimation of the layered motion, and motion-free approaches[43][42][63][65] that do not require to estimate the layered motion.

Among motion-based methods, Zhang et al.[76] proposed a method to separate each fluoroscopic image into a static layer, a slow movement layer and a fast movement layer based on the observation that different anatomical structures have various motion patterns. Similarly, Zhu et al.[78] developed a method separating each fluoroscopic image into a background layer and a coronary layer based on a Bayesian framework which combined dense motion estimation, uncertainty propagation and statistical fusion together.

In motion-free methods, the background layer and/or foreground (vessel) layer of each fluoroscopic image are modeled under certain hypotheses. Tang et al.[63] proposed an approach which was based on an assumption that the vessel layer and background layer are reconstructed from independent signals and then utilized independent componet analysis (ICA) to solve the problem. A method based on the robust principal component analysis (RPCA), which was used to detect and track the stent graft delivery device automatically in 2D fluoroscopic sequences, was proposed by D. Volpi et al.[65]. Similarly, Ma et al.[43] proposed a background modelling based approach which separates an X-ray angiogram sequence into a breathing layer, a quasi-static layer and a vessel layer using morphological closing and online robust PCA (OR-PCA) proposed by Feng et al.[20]. This method is one of the few that run online, which takes one frame as input each time and updates the background model based on the frame. Compared to its parental offline approach proposed by Ma et

[1] Haidong Hao, Hua Ma, and Theo van Walsum. **Medical Imaging 2018: Image-Guided Procedures, Robotic Interventions, and Modeling.** Vol. 10576. International Society for Optics and Photonics, 2018

al.[42], this method showed similar performance and achieved a processing rate up to 6 frames per second (fps), which is still slower than common image acquisition rates (7.5-15 fps).

The purpose of this work is separating vessel layer from X-ray angiograms. We intend to model the task as an image-to-image mapping problem, particularly, from the domain of the original X-ray angiogram to the domain of the layer that contains mainly the vessel structures.

Recently, deep learning approaches such as fully convolutional networks (FCNs) [40][50] have been developed to find such mappings. A FCN is a deep learning network architecture that has no fully connected layers. Unlike traditional convolutional neural network (CNN) for classification tasks, FCN outputs an image that has the same size as the input image(s). Typical FCNs contain a encoding path, which is similar to traditional CNNs to encode image features, and decoding path which map the learned features to pixel-wise information, e.g. a semantic segmentation map. The deep network structure consists of many convolutional layers that allows learning more powerful image representation which serves as a key to success on many object classification and segmentation tasks. Among the various FCN architectures, Noh, H. et al.[50] proposed a FCN consisting of a convolution part and a deconvolution part to segment RGB images. The convolution part extracts features from input and transforms to feature representations, whereas the deconvolution part reconstructs the object segmentation from the feature representations. Similarly, U-Net [57] connects features that are learned from the encoding path to the decoding path to facilitate feature decoding and pixel-wise information reconstruction on different scales with skip connections. This network architecture has shown exceptional performances of segmentation tasks of biomedical images [57], but it has not been explored yet to what extent it can be applied to other tasks such as vessel layer separation.

A.1.3. Overview and contributions

The purpose of this work is to develop a robust layer separation method that can run in real-time, so as to be clinically applicable. In particular, we focus on vessel layer separation directly, as this layer contains most structures of interest, such as coronary arteries, guiding catheters and guidewires. The basic idea of this work makes use of the recent advances in deep learning which has shown good performance on many computer vision tasks and medical imaging applications. Particularly, U-Net [57], a fully convolutional network was used for separating the vessel layer from X-ray angiograms, which runs online and real time. We also proposed a weight mask for each training sample by morphologically dilating the inverted gray level of the reference vessel layer to calculate the training loss and let the network focus on learning features from the vessel area.

A.2. Method

A.2.1. Network architectures

In this work, the architecture of U-Net introduced by Ronneberger, O. et al. [57] was utilized to map from original X-ray angiograms to vessel layer images. As shown in Figure A.1, firstly, the resolution of input images is 512 × 512; secondly, each convolution block has two 3 × 3 convolution layers with a ReLU (Rectified Linear Unit) activation; thirdly, max pooling with stride 2 was used between convolution blocks in the encoding path and an upconvolution layer was used to connect two successive convolution blocks in the decoding path, which consists of an upsampling operation followed by a 2 × 2 convolution without ReLU; lastly, the final read-out layer has one 1 × 1 convolution filter with a Sigmoid activation, outputting the vessel layer image with size 512 × 512.

A.2.2. Learning target

In this work, we did not follow the way in many previous works that let a network to fit to human-annotated labels. As the exact pixel values are usually not clearly defined in saliency maps, such as the vessel layer of X-ray angiogram images, which makes it difficult to obtain the “ground truth" vessel layer with human annotation, we employed the method proposed by Ma et al.[42] to generate the vessel layer as the target of our learning task. The parameter setting described by the authors was also used in our study.

A.2.3. Loss function

A data set $(\mathbf{x}, \mathbf{y})$ is employed to train the network, in which $\mathbf{x}$ is the input that is the original X-ray angiograms in this work, and $\mathbf{y}$ denotes the corresponding reference vessel layers (learning target). Let $f(\mathbf{x}, \theta)$ denote U-Net as a mapping function, in which θ is the learnable parameters of U-Net. The problem can be formulated as $\hat{\mathbf{y}} = f(\mathbf{x}, \theta)$, in which $\hat{\mathbf{y}}$ denotes the separated vessel layer output by U-Net. Then, the vessel layer separation problem requires to find the optimal parameter θ which minimizes a loss function representing the difference

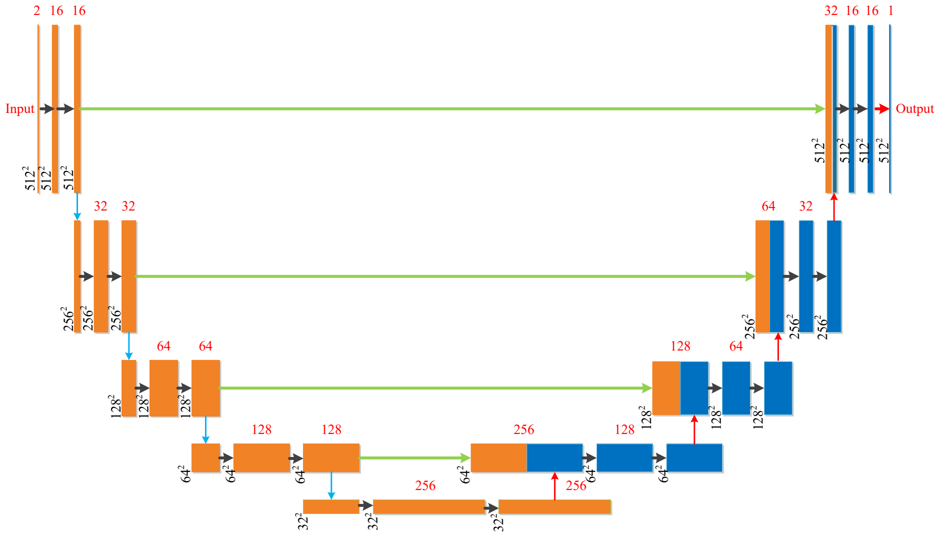

Figure A.1: U-Net architecture. Left part and right part indicate the encoding path and decoding path, respectively. Each box indicates a feature map; the number on top of each box denote the number of feature maps; the number at the lower left edge of each box is the size of corresponding feature maps; different color arrows denote different operations (black arrow: 3 × 3 convolution with a ReLU activation, blue arrow: max pooling with stride 2, green arrow: skip connection, red up arrow: upsampling operation followed by a 2 × 2 convolution without ReLU, red right arrow: 1 × 1 convolution filter with a Sigmoid activation); the orange boxes in the decoding path represent corresponding copied feature maps from the encoding path.

between $\hat{\mathbf{y}}$ and the learning target $\mathbf{y}$.

Let $\mathbf{y}$ and $\hat{\mathbf{y}}$ denote samples drawn from probability distribution $p(\mathbf{y})$ and $p(\hat{\mathbf{y}})$, respectively. The normalized pixel value of each pixel in both original angiogram and the corresponding reference image can be regarded as the probability of that pixel belonging to the background and the probability of that pixel belonging to the blood vessel is one minus the normalized pixel value[67].

To quantify the difference between the prediction and the learning target, we use binary cross entropy (BCE) shown in Equation (A.1):

$$BCE = -\frac{1}{w \times h}\sum_{i=1}^{w}\sum_{j=1}^{h}[(1-y_{i,j})\log(1-\hat{y}_{i,j}) + y_{i,j}\log\hat{y}_{i,j}] \tag{A.1}$$

in which, (w, h) is the size of the image; $y_{i,j}$ and $\hat{y}_{i,j}$ denote the probability of the pixel belonging to the background for the labeled and predicted pixel, respectively.

From the reference images in Figure A.2, it can be seen that the vessel structures possess a small area in the complete image, while the background area takes up the majority; in other words, information from the vessel and background are imbalanced. To offset the imbalance of prevalence of vessel pixels and background pixels [53] and let the network focus on learning features from the vessel area, we created a weight mask for each training sample by morphologically dilating the inverted gray level of the reference vessel layer, so as to weight the vessel pixels more. An example of weight mask is shown in Figure A.3. Using the weight mask, another loss function, weighted binary cross entropy (ωBCE) is shown in Equation (A.2):

$$\omega BCE = -\frac{1}{w \times h}\sum_{i=1}^{w}\sum_{j=1}^{h}\omega_{i,j}[(1-y_{i,j})\log(1-\hat{y}_{i,j}) + y_{i,j}\log\hat{y}_{i,j}] \tag{A.2}$$

in which, $\omega_{i,j}$ is the pixel value at location (i, j) of the weight mask.

A.2.4. Utilization of temporal information

Because X-ray angiograms are time-series data, to take the advantage of the temporal information contained in the data set, apart from the original X-ray images, we also use difference images as additional input channels. This is expected to ignore the static structures and let the network focus on moving objects. Firstly, the difference image between the current frame and its previous frame as Channel 1 (Ch1); secondly, the other

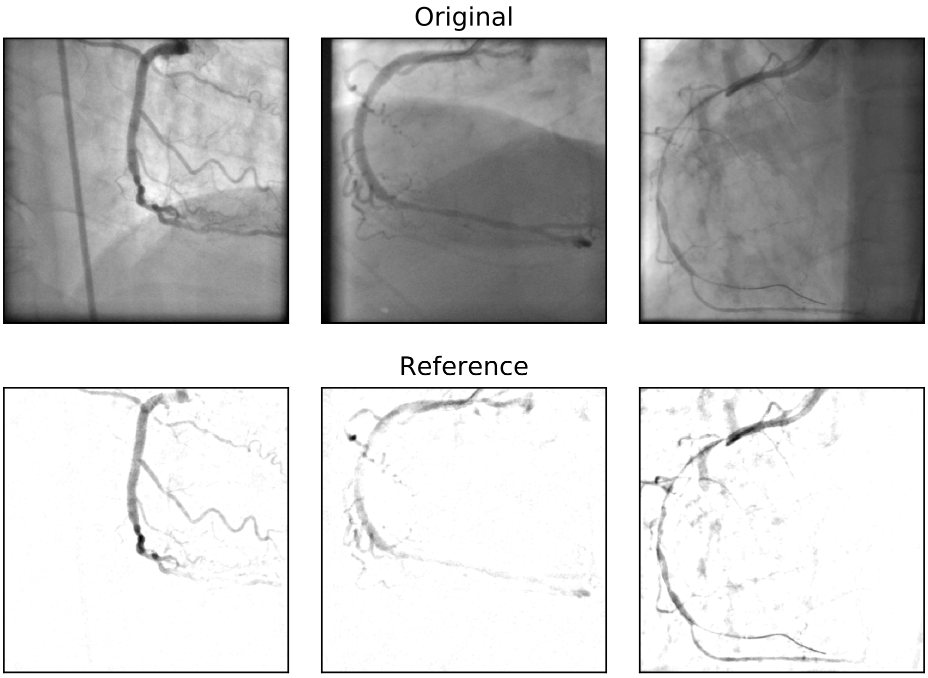

Figure A.2: Examples of original angiogram and the corresponding reference vessel layer: (First Row) the original X-ray angiogram; (Second Row) the reference vessel layer generated by the method of Ma et al. [42]

difference image which is the current frame minus the first frame as Channel 2 (Ch2); With these images, we constructed two different types of inputs for the network: the two-channel input (2Ch), which uses the original X-ray images (Channel 0) and Ch1, and the three-channel input (3Ch) with all three channels. An example of each channel input from clinical angiogram is shown in Figure A.4.

A.3. Experiments

A.3.1. Data set

Two types of data set were used in our experiments: clinical X-ray angiograms (XA) and synthetic low-contrast XA. Each dataset was divided into a training set, a validation set and a test set. The training set contains 2892 images from 26 sequences, the validating set contains 924 images from 6 sequences, and the test set contains 1068 images from 10 sequences.

X-ray angiograms The same clinical X-ray angiograms used by Ma et al.[43] were used in this work. All images were resized to 512 × 512 as the network input, and the pixel values were normalized to the range between 0 and 1.

Synthetic low-contrast X-ray angiogram Contrast agent for vessel visualization may cause kidney diseases or allergic reactions[71], so the dose of the contrast agent used in clinical application should be under control to ensure clear visualization while not causing harm to patients. To assess if our proposed method may be used to decrease contrast agent concentrations, we evaluate the performance of our methods on low-contrast data, and the method proposed by Ma et al.[43] was used to synthesize a 80% amount contrast data set and a 50% amount contrast data set. Both the two data set were preprocessed following the same procedure in section 3.1.1.

A.3.2. Evaluation metrics

To quantify the performance of vessel layer separation, the contrast-to-noise ratio (CNR) defined in Equation (A.3) was employed as one evaluation metric, which indicates the normalized difference between the average pixel value of the foreground and background. To evaluate the CNR for global and local scale, we adopted the

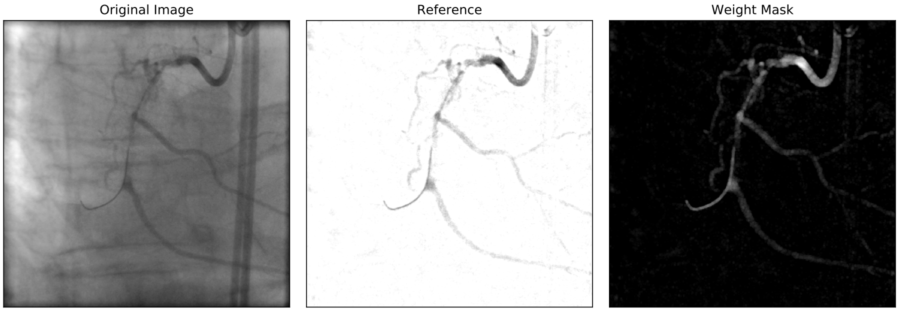

Figure A.3: An example of original angiogram, reference image, weight mask: (left) the original X-ray image; (middle) the reference vessel layer; (right) the weight mask generated from the reference image with morphological dilation; The weight mask was used to calculate the training loss.

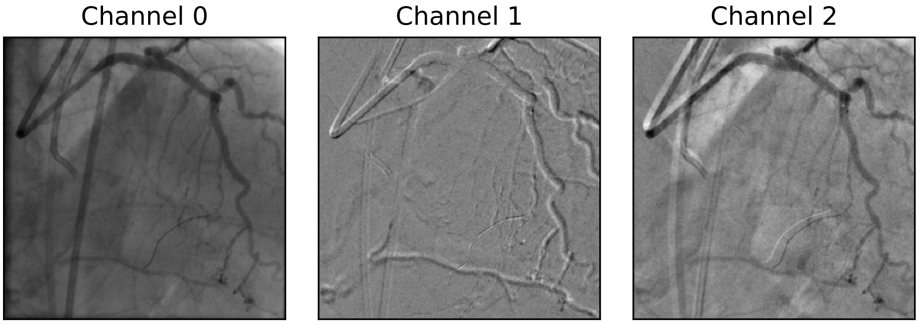

Figure A.4: An example of each channel input from clinical angiogram. (left) the current frame; (middle) the current frame minus its previous frame; (right) the current frame minus the first frame.

foreground and background masks defined in Ma et al[43], which are shown in Figure A.5.

$$CNR = \frac{|\mu_F - \mu_B|}{\sigma_B} \tag{A.3}$$

in which, the mean of foreground and background pixel values are denoted by μ_F and μ_B; the standard deviation of the background pixel values is denoted by σ_B. The global CNR measures the relation of the contrast between foreground and the whole background pixel intensities to the standard deviation of the whole background pixel intensities. On the other hand, the local CNR demonstrates the relation of the contrast between foreground and partial background surrounding the foreground pixel intensities to the standard deviation of the partial background pixel intensities.

In addition to CNR, which evaluates the contrast in a single image, we also adopt Structural SIMilarity (SSIM) proposed by Wang et al.[66] to quantify the similarity between the predicted image and the reference image, in which the luminance, contrast and structure similarities between a reference image and a predicted image were measured independently and then all the luminance, contrast and structure similarities were combined for calculating the total similarity. The definition of SSIM follows Equation (A.4).

$$SSIM = \frac{(2\mu_t\mu_p + C_1)(2\sigma_{tp} + C_2)}{(\mu_t^2 + \mu_p^2 + C_1)(\sigma_t^2 + \sigma_p^2 + C_2)} \tag{A.4}$$

in which, μ_t and μ_p are the means of the reference image and the corresponding prediction image, respectively; σ_t^2 and σ_p^2 are the corresponding variances; σ_{tp} is the covariance between the reference image and the corresponding prediction image; $C_1 = (K_1L)^2$ and $C_2 = (K_2L)^2$, where $K_1 = 0.01$, $K_2 = 0.03$ proposed by Wang et al.[66] were adopted here, and L is the range of the pixel value, i.e. 1 in our work. Mean SSIM (MSSIM) is the

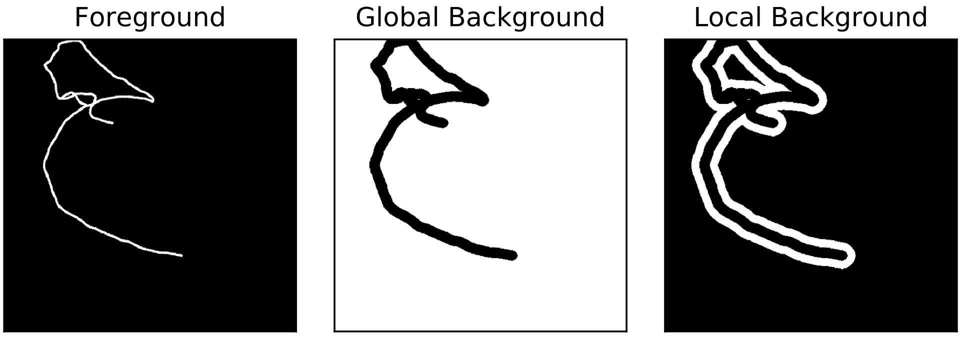

Figure A.5: An example of foreground and background proposed by Ma et al.[43]. (left) foreground (white area); (middle) global background (white area); (right) local background (white area).

average SSIM of Q pairs of images, which is defined by Equation (A.5).

$$MSSIM = \frac{1}{Q}\sum_{k=1}^{Q} SSIM_k \tag{A.5}$$

To compare the similarity between the reference image and the prediction of the network in vessel area and the whole image, we also adopted the foreground and background masks (Figure A.5) defined in Ma et al[43] to calculate the global MSSIM (the whole image) and local MSSIM (the vessel area).

A.3.3. Experiment 1: hyper parameters tuning

The optimal hyper-parameter setting needs to be found, such as learning rate, epoch number for training, network architecture, number of filters. We searched the optimal hyper-parameters by comparing the average CNR, MSSIM and prediction examples between different hyper-parameter settings in the following way: first, we tuned the hyper-parameters with ωBCE loss function using two-channel input. After doing several pilot experiments with arbitrarily chosen hyper-parameter settings, we selected a combination of these hyper parameters as a reference, which is shown in the second row of Table A.1, and then arranged four sub-experiments (Ex1-Ex4) to find the optimal learning rate, epoch number, network architecture, and number of filters of the first convolution layer. In all the sub-experiments of Table A.1, the weight mask of ωBCE loss function were generated by the corresponding reference image of the training data with the steps described as below:

Step1: Invert the pixel value of the reference image to make the vessel area with larger pixel value than the background;

Step 2: Dilate the resulted image from step 1 using a 3 × 3 square kernel;

Step 3: Normalize the pixel value of the weight mask in the range from 0 to 1.

Table A.1: Sub-experiments of Hyper parameters selection with ωBCE loss function. UNet7 is an alternative architecture of U-Net as shown in Figure A.6 and UNet9 is as shown in Figure A.1.

Experiment	**Learning rate**	**Epoch No.**	**Architecture**	**Filter No. (f)**
Reference	1e-5	50	UNet9	16
Ex1	1e-3, 5e-5, 5e-4	50	UNet9	16
Ex2	1e-5	30, 70	UNet9	16
Ex3	1e-5	50	UNet9	8, 32
Ex4	1e-5	50	UNet7	8, 16, 32

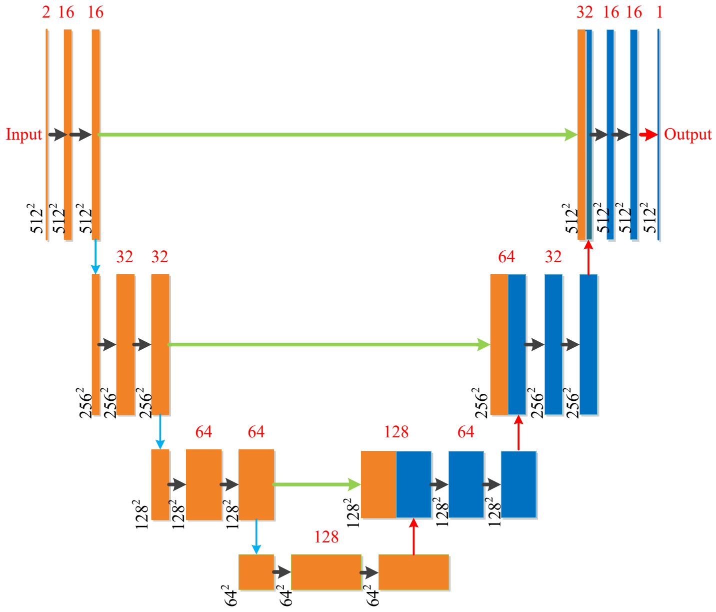

Figure A.6: An alternative architecture of U-Net.

To compare the performances between two-channel and three-channel input, after finding the best hyper parameter combination for the ωBCE loss function based on two-channel input, another sub-experiments was conducted by replacing two-channel input with three-channel input.

A.3.4. Experiment 2: Compare with other methods

The best ωBCE method was compared with one of the best method proposed by Ma et al. [43] using CNR as the evaluation metric, which is the OR-PCA method with closed-form solution (CF) as the subspace basis updating strategy and sliding window (SW) as the past information downweighting technique ($SW + CF$). The optimal parameters of $SW + CF$ method are as below: the intrinsic rank of the subspace basis $r = 5$, the regularization parameters $\lambda_1 = \lambda_2 = 2.1$ and the window size $t_0 = 3$.

A.3.5. Experiment 3: performance of low contrast data

As shown in Table A.2, to evaluate the performances of our methods on low contrast data, we replaced the clinical X-ray angiograms with a 80% amount contrast data set (80%) and a 50% amount contrast data set (50%), respectively. The reference are the same as those in Experiment 1. Because the vessels in the low contrast data are subtle, especially in the 50% amount contrast data set, we employed three-channel input to train, evaluate and test the network in addition to two-channel input.

Table A.2: Sub-experiments of Hyper parameters selection with low contrast data (50% and 80%, respectively).

Experiment	**Loss Function**	**Epoch No.**	**Data Set**
Ex5	ωBCE	50, 70	2Ch, 3Ch

A.3.6. Implementation

The network was trained and evaluated on the Dutch national supercomputer with an NVIDIA Tesla K40m GPU using Keras with Tensorflow as the backend. The network parameter θ were trained using an ADAM optimizer[36].

A.4. Results and discussion

A.4.1. Optimal hyper parameters

Three of the best hyper parameter combinations based on the ωBCE loss function are shown in Table A.3 and the corresponding average CNR and MSSIM are shown in Figure A.7 and Table A.4. For the architecture, UNet9 is similar to UNet7 in terms of CNR and MSSIM, but there are much less parameters in UNet7, so there is a trade-off between performance and speed. If the compute capability of GPUs is limited, we can choose UNet7 and get acceptable results. For the input, both 2Ch and 3Ch got similar performance. The reason may be that the vessel structure is very clear in the clinical angiograms as can be seen in Figure A.2, so the two-channel input is adequate to achieve results similar to the three-channel input. Figure A.8 illustrates two prediction examples of the three hyper parameter combinations listed in Table A.3; it also shows that the predictions of our ωBCE methods are worse than the reference image, especially in the background area. The catheter in the predictions in row 1 and the spine structures in row 2 are visible, which may be because both catheter and spine in the original angiograms are similar in both structure and colour to the vessel.

Table A.3: Three of the best hyper parameter combinations of ωBCE loss function.

Learning rate	Epoch No.	Architecture	Filter No. (f)	Dataset
1e-5	50	UNet7	16	2Ch
1e-5	50	UNet9	16	2Ch
1e-5	50	UNet9	16	3Ch

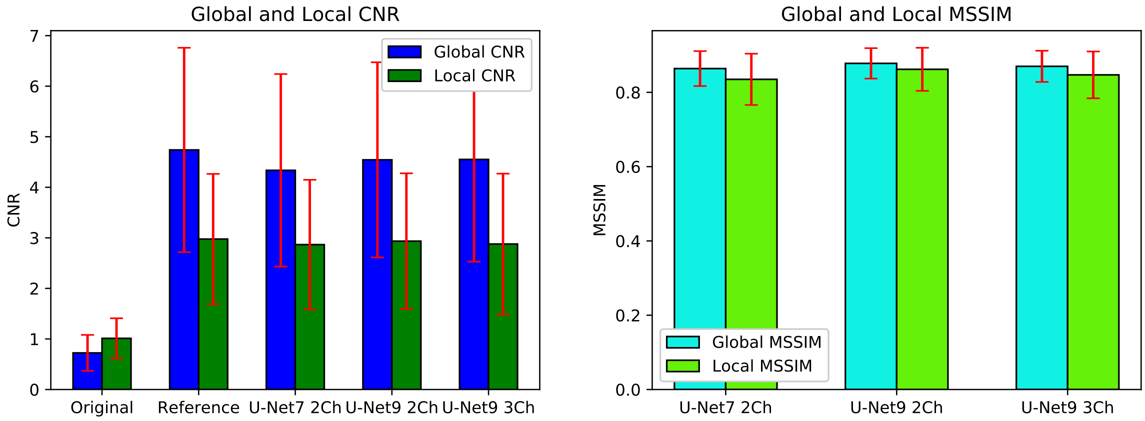

Figure A.7: Average CNR and MSSIM of various ωBCE methods

Table A.4: The average CNR and MSSIM of various ωBCE methods. (mean ± standard deviation)

Method	Local CNR	Global CNR	Local MSSIM	Global MSSIM
Reference	2.976±1.289	4.739±2.021	1	1
UNet7-2Ch	2.866±1.282	4.336±1.905	0.835±0.069	0.864±0.047
UNet9-2Ch	2.935±1.342	4.543±1.930	0.862±0.058	0.878±0.041
UNet9-3Ch	2.878±1.392	4.551±2.023	0.847±0.063	0.870±0.042

We also assessed whether the hyper parameter combinations are statistically significantly different in terms of average CNR and MSSIM, for which we employed a two-sided Wilcoxon signed-rank test[60]. The results are shown in Table A.5, in which, UNet7 is statistically significantly different from both the two UNet9 hyper parameter combinations except local CNR; the two UNet9 hyper parameter combinations are not statistically significantly different except MSSIMs.

In summary, the second row of Table A.3 (UNet9-2Ch) is the optimal hyper parameter combination among all the combinations listed in Table A.1 for our project based on ωBCE loss function.

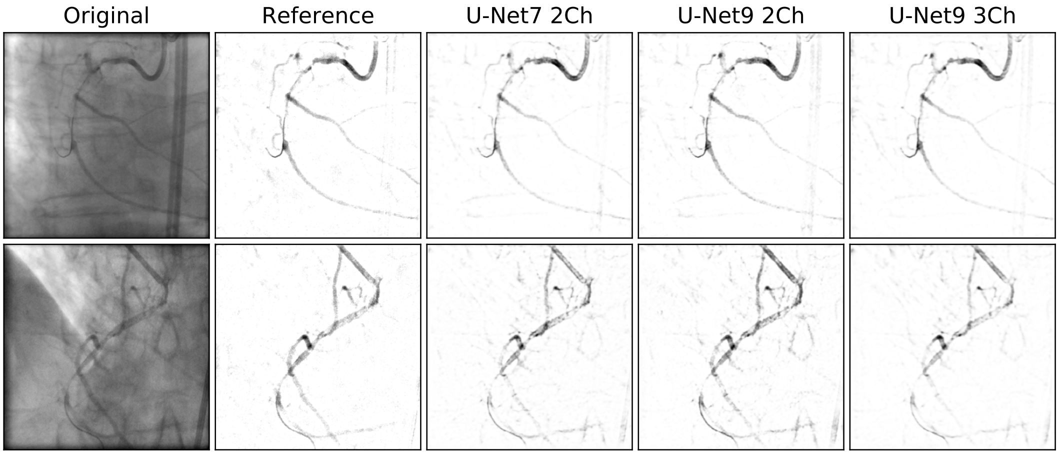

Figure A.8: Two Prediction Examples of various ωBCE methods

Table A.5: p-values among various ωBCE loss function methods in terms of CNR and MSSIM.

Method 1	Method 2	Local CNR	Global CNR	Local MSSIM	Global MSSIM
UNet7-2Ch	UNet9-2Ch	0.114	<0.001	<0.001	<0.001
UNet7-2Ch	UNet9-3Ch	0.632	0.001	0.003	0.044
UNet9-2Ch	UNet9-3Ch	0.084	0.694	0.001	0.040

A.4.2. Comparing with other methods

Figure A.10 shows examples of vessel layer separation using two different methods. Two frames (the first and second Row) from two different sequences qualitatively exhibit the performances of our methods (ωBCE) and the method $SW + CF$ presented by Ma et al[43]. These results show that the performance of our method are close to the reference image (the second column) and the method of Ma et al[43] (the last column). Compared to the work of Ma et al[43], the background obtained with our method contains fewer structures, although the vessel area seems slightly worse.

The ωBCE method has similar CNR measures to the reference image. Compared to the method of Ma et al[43], our method has superior performance on global CNR, but slightly worse on local CNR, as shown in Figure A.9 and Table A.6. This may be because there are less dark structures in the predictions of our method as shown in Figure A.10, which decrease σ_B and increase μ_B, resulting in the increase of global CNR. For local CNR, μ_F of our method are larger than that of the method of Ma et al[43], which decreases $| \mu_F - \mu_B |$, leading to the decrease of local CNR. In terms of the processing speed, the proposed method achieves a rate of about 18 fps thanks to the use of a GPU. This is faster than the common image acquisition rate in clinics (15 fps). This result demonstrates potential for a real-time clinical application.

The Wilcoxon signed-rank test [60] was done to compare the performance between our ωBCE method and the $SW + CF$ method proposed by Ma et al[43] according to 10 pairs samples. The p-values of local and global CNR are 0.185 and 0.262, respectively, so there is no statistically significant difference between the two methods in terms of local and global CNR.

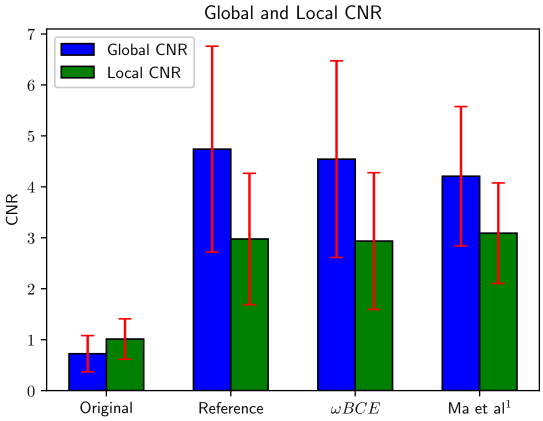

Figure A.9: Average CNR of various methods

Table A.6: Average CNR of various methods. (mean ± standard deviation)

Method	Local CNR	Global CNR
ωBCE	2.935±1.342	4.543±1.930
Ma et al^1	3.090±0.987	4.208±1.367

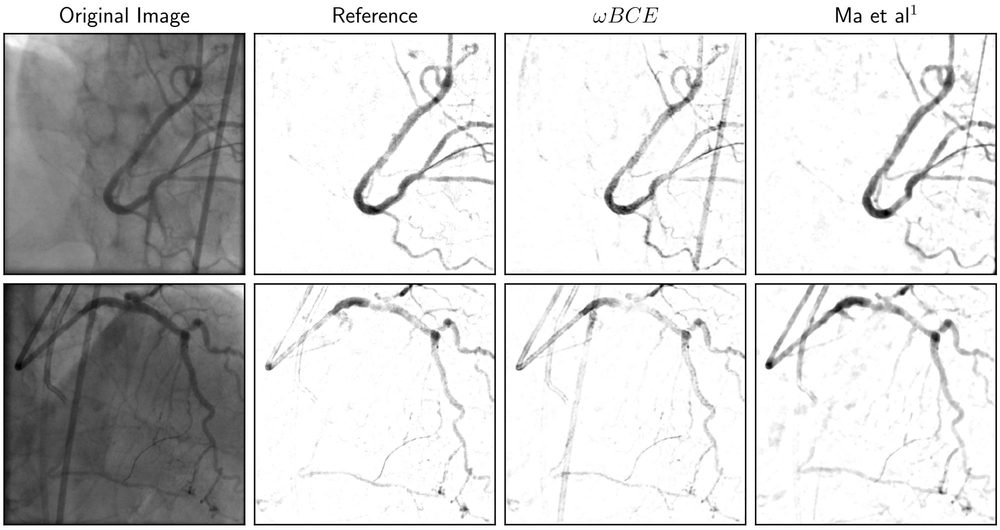

Figure A.10: Comparison of our proposed method to the method of Ma et al[43] on two examples

A.4.3. Low contrast data

Channel 1 and channel 2 shown in Figure A.4 enhance the vessel structure. We therefore utilized three-channel input to train, evaluate and test low contrast data (50% and 80%) in addition to two-channel input. The performance of three-channel input is better than that of two-channel input. The optimal hyper parameters combination for both 50% and 80% dataset based on ωBCE loss functions is shown in Table A.7.

Figure A.11 illustrates an prediction example using different contrast data based on the ωBCE method. The first, second and third row are from clinical angiogram, synthesized 80% contrast data and synthesized 50% contrast data, respectively. Columns from left to right show the original image, reference image, predictions of the optimal ωBCE method, respectively. The catheter in all the predictions was not totally removed compared to the reference image, which may be because the vessel and the catheter are similar in colour and structure, and it is difficult for the network to distinguish them. The static structures in the prediction increased as the concentration of the contrast agent decreased, which may be also because the similarities in colour and structure between vessel and static structures increased. The predictions of 80% data are nearly the same as the predictions of clinical data, although the vessel in the synthesized 80% contrast data is more subtle than that in clinical data. The reason may be that the vessel color is still different from the colour of the most static structures in 80% data. For 50% contrast data, the vessel was enhanced.

As shown in Figure A.12 (left part) and Table A.8, both clinical angiogram and low contrast data (50% and 80%)

Table A.7: The optimal hyper parameters combination for both 50% and 80% dataset of ωBCE loss function

Learning rate	Epoch No.	Architecture	Filter No. (f)	Dataset
1e-5	70	UNet9	16	3Ch

Table A.8: Average CNR and MSSIM of different dataset, the numbers 1, 0.8 and 0.5 indicate clinical angiogram, synthesized 80% contrast and 50% contrast data, respectively. (mean ± standard deviation)

Method	Local CNR	Global CNR	Local MSSIM	Global MSSIM
Reference	2.976±1.289	4.739±2.021	1	1
$\omega BCE1$	2.935±1.342	4.543±1.930	0.862±0.058	0.878±0.041
$\omega BCE0.8$	2.947±1.371	4.581±2.077	0.880±0.055	0.866±0.037
$\omega BCE0.5$	3.072±1.472	4.620±2.112	0.828±0.060	0.797±0.041

achieved nearly the same global and local CNR as the reference image, while the prediction examples in Figure A.11 are different from the reference. The reason may be that the existence of the catheter and static structures in the predictions decreases μ_B and increases σ_B simultaneously, which makes CNR nearly unchanged. Figure A.12 (right part) and Table A.8 shows the MSSIM, the local and global MSSIM of both clinical data and 80% contrast data are nearly the same, but 50% data achieved slightly low MSSIM than the other two dataset.

Table A.9: p-values between clinical angiogram, synthesized 80% contrast data and synthesized 50% contrast data using ωBCE loss function method, respectively, in terms of CNR and MSSIM.

Method 1	Method 2	Local CNR	Global CNR	Local MSSIM	Global MSSIM
$\omega BCE1$	$\omega BCE0.8$	0.182	0.211	0.0760	0.434
$\omega BCE1$	$\omega BCE0.5$	0.050	0.478	<0.001	<0.001

The Wilcoxon signed-rank test [60] was also utilized to compare the performance between clinical and low contrast dataset, the results of which are shown in Table A.9. There is no statistically significant differences between clinical angiogram and synthesized 80% contrast data, although the CNR of synthesized 80% contrast data is lower than that of clinical angiogram.

Table A.9 also shows that there are statistically significant differences between clinical angiogram and synthesized 50% contrast data in terms of both local and global MSSIM. Clinical angiogram and synthesized 50% contrast data have no statistically significant difference in terms of CNR, although the CNR of synthesized 50% contrast data is much lower than that of clinical angiogram, which indicates that ωBCE method can enhance the vessel layer.

A.5. Conclusion

We have presented a data-driven method to separate vessel layer from cardiac interventional X-ray angiograms for vessel enhancement. The method uses a fully convolutional network to map the original X-ray image to a vessel layer image in which vessel structures have better visibility. We trained the network with automatically generated images of the vessel structure, the experimental results show that our proposed method is able to compute the vessel layer and enhance vessel structures. The proposed method shows an improved CNR compared to the original X-ray images, and has a performance that is similar to the state-of-the-art method. As the proposed method has a processing rate of about 18 frames per second, it has potential for real-time clinical application. We also investigated the performance of our method on low contrast dataset and the performance on the 80% contrast dataset is nearly the same as the clinical angiograms, which indicates a potential to reduce the dose of contrast agent in coronary interventions.

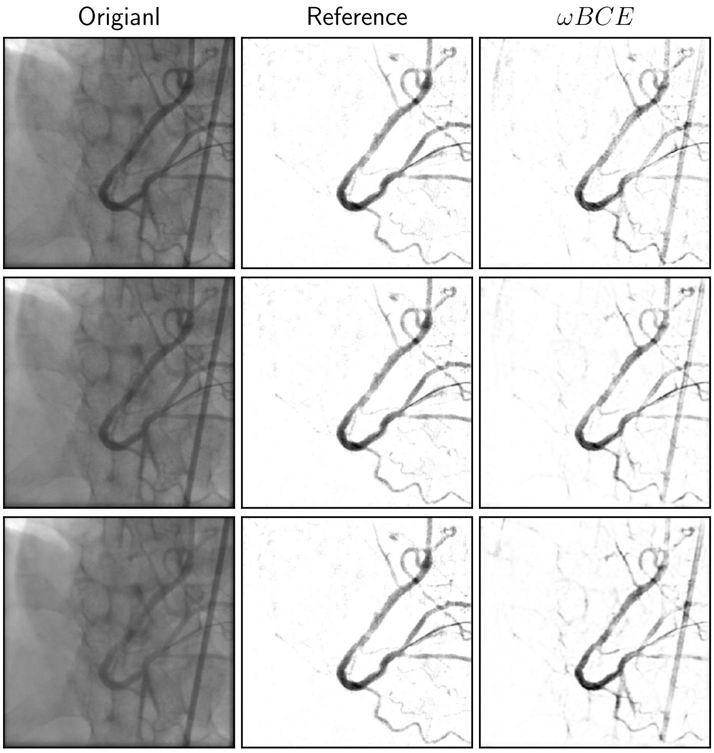

Figure A.11: A prediction example of different data with respective optimal methods, 1^{st} Row: Clinical angiogram; 2^{nd} Row: synthesized 80% contrast data; 3^{rd} Row: synthesized 50% contrast data; 3^{rd} Column: ωBCE loss function.

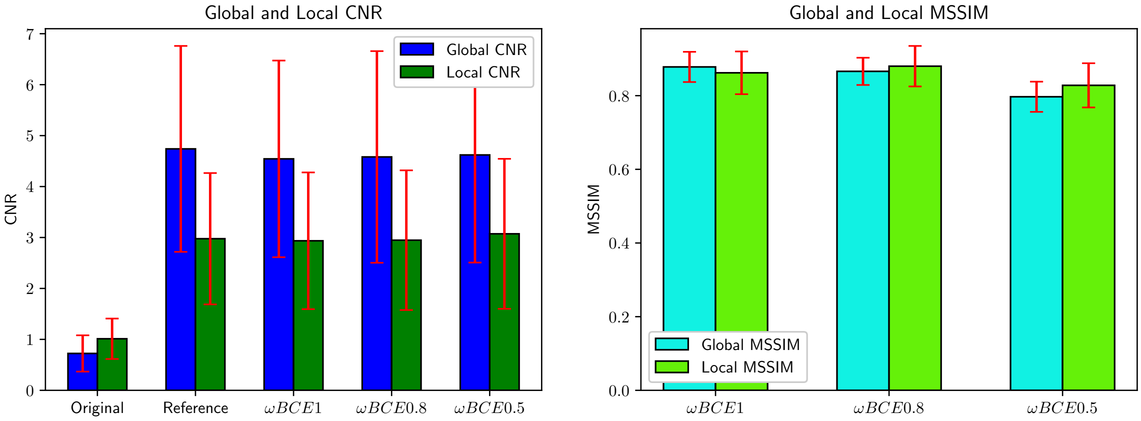

Figure A.12: Average CNR and MSSIM of different dataset, the numbers 1, 0.8 and 0.5 indicate clinical angiogram, synthesized 80% contrast and 50% contrast data, respectively.

Bibliography

[1] Coronary heart disease. https://www.nhlbi.nih.gov/health-topics/coronary-heart-disease#Treatment.

[2] Pci - percutaneous coronary intervention. http://dxline.info/diseases/pci-percutaneous-coronary-intervention#prettyPhoto.

[3] Interventional x-ray solutions. https://www.usa.philips.com/healthcare/solutions/interventional-xray.

[4] Data science: Machine learning vs. rules based systems. https://www.forbes.com/sites/teradata/2015/12/15/data-science-machine-learning-vs-rules-based-systems/#52aef7a02119, 2015.

[5] Machine learning vs rules systems. https://deparkes.co.uk/2017/11/24/machine-learning-vs-rules-systems/, 2017.

[6] The top 10 causes of death. http://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death, 2018.

[7] Shadi Albarqouni, Javad Fotouhi, and Nassir Navab. X-ray in-depth decomposition: Revealing the latent structures. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 444–452. Springer, 2017.

[8] Scott E Reed andHonglak Lee. Training deep neural networks on noisy labels with bootstrapping. In *Accepted as a workshop contribution at ICLR*, pages 1–11, 2015.

[9] Vincent Auvray, Patrick Bouthemy, and Jean Liénard. Jointmotion estimation and layer segmentation in transparent image sequenc: application to noise reduction in X-ray image sequences. *EURASIP Journal on Advances in Signal Processing*, 2009:19, 2009.

[10] Nora Baka, CT Metz, Carl Schultz, Lisan Neefjes, Robert Jan van Geuns, Boudewijn PF Lelieveldt, Wiro J Niessen, Theo van Walsum, and Marleen de Bruijne. Statistical coronary motion models for 2D+ t/3D registration of X-ray coronary angiography and CTA. *Medical Image Analysis*, 17(6):698–709, 2013.

[11] Nora Baka, CT Metz, Carl J Schultz, R-J van Geuns, Wiro J Niessen, and Theo van Walsum. Oriented Gaussian mixture models for nonrigid 2D/3D coronary artery registration. *Medical Imaging, IEEE Transactions on*, 33(5):1023–1034, 2014.

[12] Neslihan Bayramoglu, Mika Kaakinen, Lauri Eklund, and Janne Heikkila. Towards virtual H&E staining of hyperspectral lung histology images using conditional generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 64–71, 2017.

[13] Sujata K Bhatia. *Biomaterials for clinical applications*. Springer Science & Business Media, 2010.

[14] Y-Lan Boureau, Jean Ponce, and Yann LeCun. A theoretical analysis of feature pooling in visual recognition. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 111–118, 2010.

[15] Wonmin Byeon. Image analysis with long short-term memory recurrent neural networks. 2016.

[16] Agisilaos Chartsias, Thomas Joyce, Rohan Dharmakumar, and Sotirios A Tsaftaris. Adversarial image synthesis for unpaired multi-modal cardiac data. In *International Workshop on Simulation and Synthesis in Medical Imaging*, pages 3–13. Springer, 2017.

[17] Jianxu Chen, Lin Yang, Yizhe Zhang, Mark Alber, and Danny Z Chen. Combining fully convolutional and recurrent neural networks for 3d biomedical image segmentation. In *Advances in Neural Information Processing Systems*, pages 3036–3044, 2016.

[18] Robert A Close, Craig K Abbey, Craig A Morioka, and James S Whiting. Accuracy assessment of layer decomposition using simulated angiographic image sequences. *Medical Imaging, IEEE Transactions on*, 20(10):990–998, 2001.

[19] Pedro Costa, Adrian Galdran, Maria Ines Meyer, Meindert Niemeijer, Michael Abràmoff, Ana Maria Mendonça, and Aurélio Campilho. End-to-end adversarial retinal image synthesis. *IEEE transactions on medical imaging*, 37(3):781–791, 2018.

[20] Jiashi Feng, Huan Xu, and Shuicheng Yan. Online robust pca via stochastic optimization. In *Advances in Neural Information Processing Systems*, pages 404–412, 2013.

[21] Peter Fischer, Thomas Pohl, Andreas Maier, and Joachim Hornegger. Surrogate-driven estimation of respiratory motion and layers in X-ray fluoroscopy. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 282–289. Springer, 2015.

[22] Alejandro F Frangi, Wiro J Niessen, Koen L Vincken, and Max A Viergever. Multiscale vessel enhancement filtering. In *Medical Image Computing and Computer-Assisted Intervention*, pages 130–137. Springer, 1998.

[23] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[24] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

[25] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

[26] Haidong Hao, Hua Ma, and Theo van Walsum. Vessel layer separation in X-ray angiograms with fully convolutional network. In *Proc.SPIE 10576, Medical Imaging 2018: Image-Guided Procedures, Robotic Interventions, and Modeling*, page 105761X, 2018.

[27] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015.

[28] Yipeng Hu, Eli Gibson, Li-Lin Lee, Weidi Xie, Dean C Barratt, Tom Vercauteren, and J Alison Noble. Freehand ultrasound image simulation with spatially-conditioned generative adversarial networks. In *Molecular Imaging, Reconstruction and Analysis of Moving Body Organs, and Stroke Imaging and Treatment*, pages 105–115. Springer, 2017.

[29] Yipeng Hu, Eli Gibson, Tom Vercauteren, Hashim U Ahmed, Mark Emberton, Caroline M Moore, J Alison Noble, and Dean C Barratt. Intraoperative organ motion models with an ensemble of conditional generative adversarial networks. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 368–376. Springer, 2017.

[30] Aapo Hyvärinen and Erkki Oja. Independent component analysis: algorithms and applications. *Neural networks*, 13(4-5):411–430, 2000.

[31] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[32] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. *CVPR*, 2017.

[33] Mingxin Jin, Rong Li, Jian Jiang, and Binjie Qin. Extracting contrast-filled vessels in X-ray angiography by graduated RPCA with motion coherency constraint. *Pattern Recognition*, 63:653–666, 2017.

[34] Ishan Jindal, Matthew Nokleby, and Xuewen Chen. Learning deep networks from noisy labels with dropout regularization. In *Data Mining (ICDM), 2016 IEEE 16th International Conference on*, pages 967–972. IEEE, 2016.

[35] Morton J Kern, Michael J Lim, and Paul Sorajja. *The Interventional Cardiac Catheterization Handbook E-Book*. Elsevier Health Sciences, 2017.

[36] Diederik Kingma and Jimmy Ba. ADAM: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[37] David G Kleinbaum and Mitchel Klein. *Logistic regression: a self-learning text*. Springer Science & Business Media, 2010.

[38] Yong Geun Lee, Jeongjin Lee, Yeong-Gil Shin, and Ho Chul Kang. Low-dose 2d x-ray angiography enhancement using 2-axis pca for the preservation of blood-vessel region and noise minimization. *Computer methods and programs in biomedicine*, 123:15–26, 2016.

[39] Geert Litjens, Thijs Kooi, Babak Ehteshami Bejnordi, Arnaud Arindra Adiyoso Setio, Francesco Ciompi, Mohsen Ghafoorian, Jeroen AWM van der Laak, Bram van Ginneken, and Clara I Sánchez. A survey on deep learning in medical image analysis. *Medical image analysis*, 42:60–88, 2017.

[40] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015.

[41] Cristian Lorenz, I-C Carlsen, Thorsten M Buzug, Carola Fassnacht, and Jürgen Weese. Multi-scale line segmentation with automatic estimation of width, contrast and tangential direction in 2d and 3d medical images. In *CVRMed-MRCAS'97*, pages 233–242. Springer, 1997.

[42] Hua Ma, Gerardo Dibildox, Jyotirmoy Banerjee, Wiro Niessen, Carl Schultz, Evelyn Regar, and Theo van Walsum. Layer separation for vessel enhancement in interventional X-ray angiograms using morphological filtering and robust PCA. In *Workshop on Augmented Environments for Computer-Assisted Interventions*, pages 104–113. Springer, 2015.

[43] Hua Ma, Ayla Hoogendoorn, Evelyn Regar, Wiro J Niessen, and Theo van Walsum. Automatic online layer separation for vessel enhancement in X-ray angiograms for percutaneous coronary interventions. *Medical Image Analysis*, 39:145–161, 2017.

[44] Andrew L Maas, Awni Y Hannun, and Andrew Y Ng. Rectifier nonlinearities improve neural network acoustic models. In *Proc. icml*, volume 30, page 3, 2013.

[45] Davide Maccagni, Cosmo Godino, Azeem Latib, Lorenzo Azzalini, Vittorio Pazzanese, Alaide Chieffo, Alberto Margonato, and Antonio Colombo. Analysis of a low dose protocol to reduce patient radiation exposure during percutaneous coronary interventions. *American journal of cardiology*, 119(2):203–209, 2017.

[46] Pim Moeskops, Mitko Veta, Maxime W Lafarge, Koen AJ Eppenhof, and Josien PW Pluim. Adversarial training and dilated convolutions for brain MRI segmentation. In *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, pages 56–64. Springer, 2017.

[47] E Nasr-Esfahani, N Karimi, MH Jafari, SMR Soroushmehr, S Samavi, BK Nallamothu, and K Najarian. Segmentation of vessels in angiograms using convolutional neural networks. *Biomedical Signal Processing and Control*, 40:240–251, 2018.

[48] Ebrahim Nasr-Esfahani, Shadrokh Samavi, Nader Karimi, SM Reza Soroushmehr, Kevin Ward, Mohammad H Jafari, Banafsheh Felfeliyan, B Nallamothu, and Kayvan Najarian. Vessel extraction in X-ray angiograms using deep learning. In *Engineering in Medicine and Biology Society (EMBC), 2016 IEEE 38th Annual International Conference of the*, pages 643–646. IEEE, 2016.

[49] Dong Nie, Roger Trullo, Jun Lian, Caroline Petitjean, Su Ruan, Qian Wang, and Dinggang Shen. Medical image synthesis with context-aware generative adversarial networks. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 417–425. Springer, 2017.

[50] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning deconvolution network for semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1520–1528, 2015.

[51] Sílvia Delgado Olabarriaga, M Breeuwer, and WJ Niessen. Evaluation of hessian-based filters to enhance the axis of coronary arteries in ct images. In *International Congress Series*, volume 1256, pages 1191–1196. Elsevier, 2003.

[52] Maria Panayiotou, Andrew P King, R James Housden, YingLiang Ma, Michael Cooklin, Mark O'Neill, Jaswinder Gill, C Aldo Rinaldi, and Kawal S Rhode. A statistical method for retrospective cardiac and respiratory motion gating of interventional cardiac X-ray images. *Medical Physics*, 41(7):071901, 2014.

[53] Jay Patravali, Shubham Jain, and Sasank Chilamkurthy. 2D-3D fully convolutional neural networks for cardiac MR segmentation. *arXiv preprint arXiv:1707.09813*, 2017.

[54] Peter Peduzzi, John Concato, Elizabeth Kemper, Theodore R Holford, and Alvan R Feinstein. A simulation study of the number of events per variable in logistic regression analysis. *Journal of clinical epidemiology*, 49(12):1373–1379, 1996.

[55] J Samuel Preston, Caleb Rottman, Arvidas Cheryauka, Larry Anderton, Ross T Whitaker, and Sarang C Joshi. Multi-layer deformation estimation for fluoroscopic imaging. In *Information Processing in Medical Imaging*, pages 123–134, 2013.

[56] David Rivest-Henault, Hari Sundar, and Mohamed Cheriet. Nonrigid 2D/3D registration of coronary artery models with live fluoroscopy for guidance of cardiac interventions. *Medical Imaging, IEEE Transactions on*, 31(8):1557–1572, 2012.

[57] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 234–241. Springer, 2015.

[58] Yoshinobu Sato, Shin Nakajima, Hideki Atsumi, Thomas Koller, Guido Gerig, Shigeyuki Yoshida, and Ron Kikinis. 3d multi-scale line filter for segmentation and visualization of curvilinear structures in medical images. In *CVRMed-MRCAS'97*, pages 213–222. Springer, 1997.

[59] Matthias Schneider and Hari Sundar. Automatic global vessel segmentation and catheter removal using local geometry information and vector field integration. In *Biomedical Imaging: From Nano to Macro, 2010 IEEE International Symposium on*, pages 45–48. IEEE, 2010.

[60] Sidney Siegal. *Nonparametric statistics for the behavioral sciences.* McGraw-hill, 1956.

[61] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1): 1929–1958, 2014.

[62] Sainbayar Sukhbaatar, Joan Bruna, Manohar Paluri, Lubomir Bourdev, and Rob Fergus. Training convolutional networks with noisy labels. *arXiv preprint arXiv:1406.2080*, 2014.

[63] Songyuan Tang, Yongtian Wang, and Yen-Wei Chen. Application of ICA to X-ray coronary digital subtraction angiography. *Neurocomputing*, 79:168–172, 2012.

[64] Lucas Theis, Aäron van den Oord, and Matthias Bethge. A note on the evaluation of generative models. *arXiv preprint arXiv:1511.01844*, 2015.

[65] Daniele Volpi, Mhd H Sarhan, Reza Ghotbi, Nassir Navab, Diana Mateus, and Stefanie Demirci. Online tracking of interventional devices for endovascular aortic repair. *International journal of computer assisted radiology and surgery*, 10(6):773–781, 2015.

[66] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[67] Yunchao Wei, Xiaodan Liang, Yunpeng Chen, Xiaohui Shen, Ming-Ming Cheng, Jiashi Feng, Yao Zhao, and Shuicheng Yan. STC: A simple to complex framework for weakly-supervised semantic segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 2016.

[68] Jelmer M Wolterink, Anna M Dinkla, Mark HF Savenije, Peter R Seevinck, Cornelis AT van den Berg, and Ivana Išgum. Deep MR to CT synthesis using unpaired data. In *International Workshop on Simulation and Synthesis in Medical Imaging*, pages 14–23. Springer, 2017.

[69] Jelmer M Wolterink, Tim Leiner, Max A Viergever, and Ivana Išgum. Generative adversarial networks for noise reduction in low-dose CT. *IEEE transactions on medical imaging*, 36(12):2536–2545, 2017.

[70] Xianliang Wu, James Housden, YingLiang Ma, Benjamin Razavi, Kawal Rhode, and Daniel Rueckert. Fast catheter segmentation from echocardiographic sequences based on segmentation from corresponding X-ray fluoroscopy for cardiac catheterization interventions. *IEEE transactions on medical imaging*, 34(4): 861–876, 2015.

[71] Diane K Wysowski and Parivash Nourjah. Deaths attributed to X-ray contrast media on US death certificates. *American journal of roentgenology*, 186(3):613–615, 2006.

[72] SHI Xingjian, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In *Advances in neural information processing systems*, pages 802–810, 2015.

[73] Dong Yang, Daguang Xu, S Kevin Zhou, Bogdan Georgescu, Mingqing Chen, Sasa Grbic, Dimitris Metaxas, and Dorin Comaniciu. Automatic liver segmentation using an adversarial image-to-image network. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 507–515. Springer, 2017.

[74] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.

[75] Matthew D Zeiler, Dilip Krishnan, Graham W Taylor, and Rob Fergus. Deconvolutional networks. 2010.

[76] Wei Zhang, Haibin Ling, Simone Prummer, Kevin Shaohua Zhou, Martin Ostermeier, and Dorin Comaniciu. Coronary tree extraction using motion layer separation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 116–123. Springer, 2009.

[77] Yizhe Zhang, Lin Yang, Jianxu Chen, Maridel Fredericksen, David P Hughes, and Danny Z Chen. Deep adversarial networks for biomedical image segmentation utilizing unannotated images. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 408–416. Springer, 2017.

[78] Ying Zhu, Simone Prummer, Peng Wang, Terrence Chen, Dorin Comaniciu, and Martin Ostermeier. Dynamic layer separation for coronary DSA and enhancement in fluoroscopic sequences. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 877–884. Springer, 2009.