Layer Separation in X-ray Angiograms for Vessel Enhancement with Fully Convolutional Network

Haidong Hao[1], Hua Ma[2], and Theo van Walsum[2]

[1] Faculty of EEMCS, Delft University of Technology, Delft, Netherlands
[2] Biomedical Imaging Group Rotterdam, Erasmus MC, Rotterdam, Netherlands
h.ma@erasmusmc.nl

Abstract. Percutaneous coronary intervention is a treatment for coronary artery disease, which is performed under image-guidance using X-ray angiography. The intensities in an X-ray image are a superimposition of 2D structures projected from 3D anatomical structures, which makes robust information processing challenging. The purpose of this work is to investigate to what extent vessel layer separation can be achieved with deep learning, especially adversarial networks. To this end, we develop and evaluate a deep learning based method for vessel layer separation. In particular, the method utilizes a fully convolutional network (FCN), which was trained by two different strategies: an L_1 loss and a combination of L_1 and adversarial losses. The experiment results show that the FCN trained with both losses can well enhance vessel structures by separating the vessel layer, while the L_1 loss results in better contrast. In contrast to traditional layer separation methods [1], both our methods can be executed much faster and thus have potential for real-time applications.

1 Introduction

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 3D anatomical structures. The overlapping nature of structures in X-ray angiograms (XA) makes robust information processing challenging. Layer separation was proposed for separating 2D overlapping structures in XA and putting them in different layers by exploiting temporal information. As a result, structures with similar motion patterns or appearances are grouped together and ready for further analysis without interference of structures in other layers [1].

In contrast to traditional methods [1], methods based on machine learning, particularly deep learning, have been reported to gain excellent performance in solving medical imaging problems [3], including layer separation in XA [4].

In this scenario, layer separation is viewed as an image-to-image translation problem, in which a mapping function is learned to translate an input (XA) to an output (vessel layers). Performance of image-to-image translation may be further boosted by generative adversarial networks (GANs, [6]). GANs consist of two networks, a generator and a discriminator. The idea of adversarial training has been applied and achieved good performance in solving medical imaging problems [7]. Nevertheless, to what extent it can be used for layer separation for vessel enhancement in XA has not been explored yet.

In this paper, we investigate and evaluate deep learning based layer separation methods for vessel enhancement in XA, including trained by adversarial networks ($AN + L_1$ method) introduced in [10] and a conventional L_1 loss (L_1 method). In particular, the work focuses on transforming the XA 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 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 [4]; 3) assessing the proposed methods for low-contrast scenarios with synthetic XA data, which show robust performance.

2 Method

While the original GAN [6] generates new samples from random noise z, we adopt the approach introduced in [10] that trains a generator to generate a new image y from the input image x and a random noise z. Different from [10], 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 $G : x \rightarrow y$, where x is an input XA and y represents the desired output vessel layer. The method overview is illustrated in Fig. 1.

2.1 Training objective

The GAN objective of our approach can be described as Eq. 1,

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

where G is the generator, D is the discriminator, x and y denote the input XA and the reference vessel layer, respectively. Note that $\mathcal{L}_{GAN}(G, D)$ 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 ([10]) or L_2 ([8]) distance, penalizing the generator output being different from the reference. We choose the L_1 distance (see Eq. 2) for our approach, as it preserves finer details in the images than L_2, which is advantageous to small vessel enhancement.

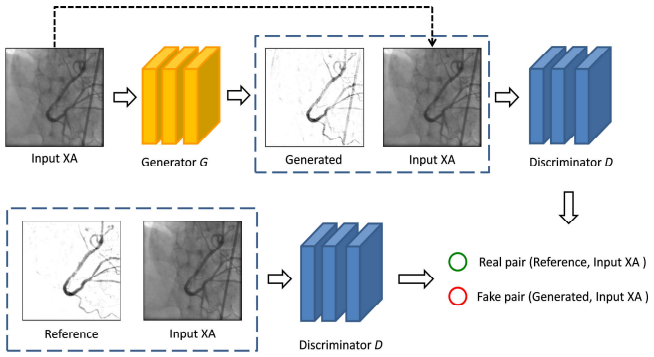

Fig. 1: Overview of our approach. The generator G learns a pixel-to-pixel transformation that maps an input XA to a vessel layer where the vessel structure is enhanced and the background structures are removed. The Discriminator D receives the input XA and the vessel layer as an input pair. D is trained 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.

$$\mathcal{L}_{L_1}(G) = E_{x,y\sim p_{data}(x,y)}||y - G(x)||_1 \tag{2}$$

The total objective of our approach is expressed in Eq. 3, where λ is a weight balancing the two terms.

$$\min_G \max_D \mathcal{L}_{GAN}(G,D) + \lambda \mathcal{L}_{L_1}(G) \tag{3}$$

2.2 Generator G

We used a U-Net-like architecture [5] for G, slightly modified from our previous work [4]. First, batch normalization [9] 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 [10] to reduce overfitting. The generator architecture can be referred to Fig. 2.

As XA sequences are time series of images, temporal information between frames is useful for distinguishing foreground and background structures. 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

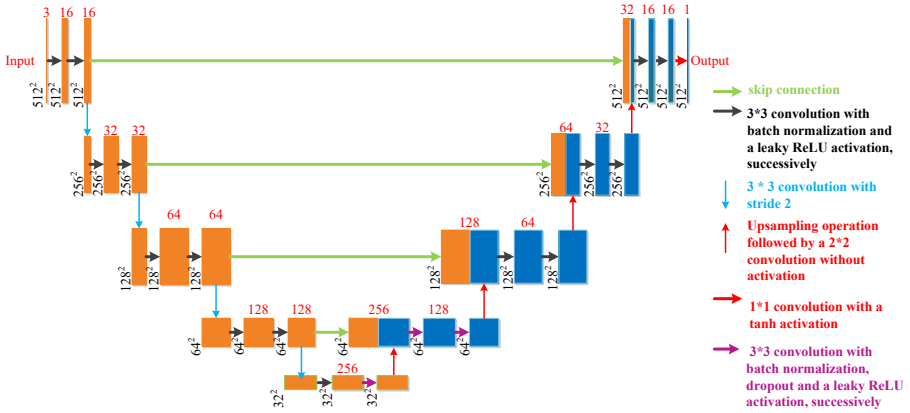

Fig. 2: Generator architecture. The left and right sides are an encoder and a decoder, respectively. Each box denotes a feature map; the number on top of each box indicates the number of feature maps; the number at the lower left edge of each box is the size of corresponding feature maps; the orange boxes in the decoder represent corresponding copied feature maps from the encoder.

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.

According to Eq. 2, training G (also D according to Eq. 1) requires a training reference y. To create the training reference, we used the layer separation approach in [2] to generate the "ground truth" vessel layer. The pixel values of the resulted vessel layer were then normalized to the range from -1 to 1 due to the last activation layer of G (see Section 2.2).

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 five 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.

3 Experiment and Result

3.1 Data

42 XA sequences were acquired with Siemens AXIOM-Artis biplane system from 21 patients undergoing a PCI procedure in the Erasmus MC, Rotterdam. The

frame rate is 15 fps. The 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 sets with nearly the same frame numbers for cross-validation.

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 512×512 so that input images to the neural networks are of the same dimension; 2) the pixel values of all images were normalized to the range from -1 to 1.

3.2 Evaluation metrics

After normalizing the range of references and predictions from 0 to 1, we evaluate the quality of the vessel layer images using contrast-to-noise ratio (CNR). As CNR is a metric based on only the output of G, we additionally used structure similarity (SSIM), following the settings from [11] to measure the similarity between the generator output and the reference. The CNR and SSIM were computed in both local and global scale using the mask images defined in [1]. 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.

3.3 Implementation

All the networks were trained and evaluated on SURFsara 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 [12].

3.4 Experiment 1: Evaluation on clinical XA

We compared the performance of training the generator with L_1 only (Eq. 2) and the combination of L_1 and adversarial loss (Eq. 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 with the method presented in [4] using average CNR and SSIM of 42 frames from 4 sequence. The optimal hyper-parameters obtained from cross-validation for both $AN + L_1$ and L_1 methods are (input = 2Ch, learning rate for G = 5×10^{-4} , learning rate for D = 5×10^{-4} , epoch number = 50, $\lambda = 10$) and (input = 2Ch, learning rate for G = 5×10^{-4} , epoch number = 50), respectively.

Average CNR and SSIM of both our proposed methods and the method proposed in [4] based on the test data of clinical XA are shown in Fig. 3. Fig. 5 illustrates two prediction examples of these methods. As shown in Fig. 3, all the three methods achieve nearly the same local CNR that is also similar to the reference. Fig. 5 shows that vessel area of the $AN + L_1$ method is the brightest; 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. $AN + L_1$ method is statistically different from L_1 method; $AN + L_1$ method and the method proposed in [4] are also statistically different except for local CNR; differences between L_1 method and the method proposed in [4] are only statistically significant for SSIM.

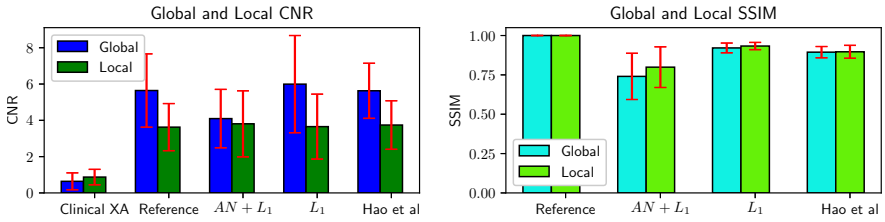

Fig. 3: Average CNR and SSIM of various methods based on clinical XA.

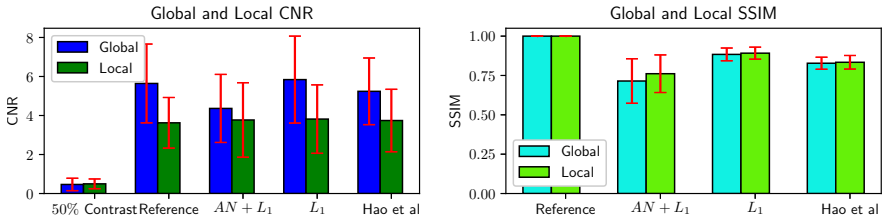

Fig. 4: Average CNR and SSIM of various methods based on low-contrast XA.

3.5 Experiment 2: Evaluation on synthetic low-contrast XA

According to [1], layer separation has the potential to enhance vessels in XA 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, we also evaluated our proposed method on low-contrast XA synthesized from the clinical XA, with the same references as those in Exp. 1. We also examined the influences of the input channels and compared to the method presented in [4]. The synthetic images simulate a 50% lower contrast concentration and were constructed using an offline robust principal component analysis (RPCA) approach [1].

The optimal hyper-parameters for both $AN + L_1$ and L_1 methods based on low-contrast XA are the same as those of the clinical XA. Average CNR and SSIM of our proposed methods and the method presented in [4] based on the test data of low contrast XA are shown in Fig. 4. Fig. 6 illustrates two prediction examples of these methods. As illustrated in Fig. 4, all the three methods achieve nearly the same local CNR that is also similar to the reference.

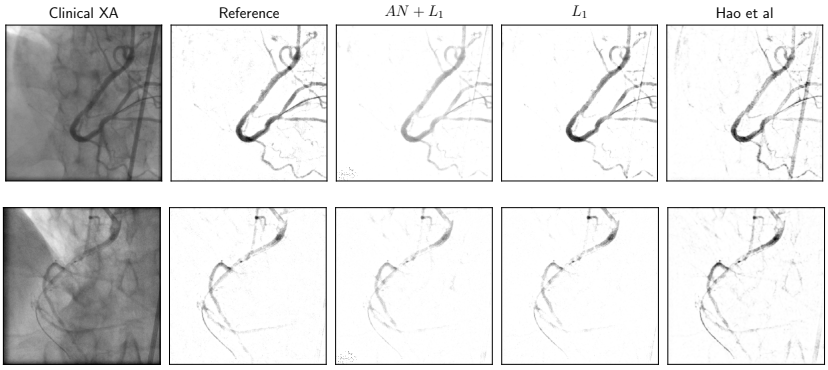

Fig. 5: Two prediction examples of various methods based on clinical XA.

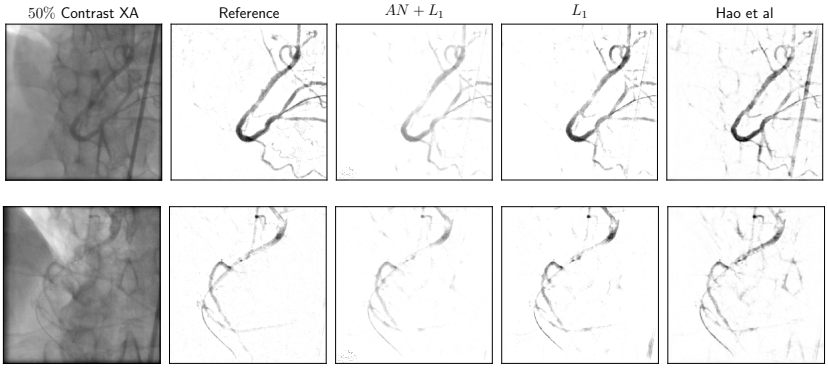

Fig. 6: Two prediction examples of various methods based on low-contrast XA.

4 Discussion

In summary, the performance of $AN + L_1$ method is slightly worse than that of the L_1 method based on the test data of both clinical XA and low-contrast XA. This may be because $AN + L_1$ 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 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.

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 decreases the global CNR. In Fig. 5 and 6, the vessel area of both the L_1 method and the method proposed in [4] 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.

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 previous 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.

References

1. Hua Ma et al. Automatic online layer separation for vessel enhancement in X-ray angiograms for percutaneous coronary interventions. Med Image Anal, 39:145–161, 2017.
2. Hua Ma et al. Layer separation for vessel enhancement in interventional X-ray angiograms using morphological filtering and robust PCA. In Workshop on AE-CAI, pages 104–113. Springer, 2015.
3. Geert Litjens et al. A survey on deep learning in medical image analysis. Med Image Anal, 42:60–88, 2017.
4. Haidong Hao et al. 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, 2018.
5. Olaf Ronneberger et al. U-net: Convolutional networks for biomedical image segmentation. In Int. Conf. on MICCAI, pages 234–241. Springer, 2015.
6. Ian Goodfellow et al. Generative adversarial nets. In Advances in neural information processing systems, pages 2672–2680, 2014.
7. Dong Nie et al. Medical image synthesis with context-aware generative adversarial networks. In Int. Conf. on MICCAI, pages 417–425. Springer, 2017.
8. Jelmer M Wolterink et al. Generative adversarial networks for noise reduction in low-dose CT. IEEE T-MI, 36(12):2536–2545, 2017.
9. Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167, 2015.
10. Phillip Isola et al. Image-to-image translation with conditional adversarial networks. CVPR, 2017.
11. Zhou Wang et al. Image quality assessment: from error visibility to structural similarity. IEEE T IMAGE PROCESS, 13(4):600–612, 2004.
12. Diederik Kingma and Jimmy Ba. ADAM: A method for stochastic optimization. arXiv preprint arXiv:1412.6980, 2014.