Medical Image Enhancement using Super Resolution Methods

Koki Yamashita and Konstantin Markov[0000−0003−1838−4789]

University of Aizu, Fukushima 965-8580, Japan
{m5231120,markov}@u-aizu.ac.jp

Abstract. Deep Learning image processing methods are gradually gaining popularity in a number of areas including medical imaging. Classification, segmentation, and denoising of images are some of the most demanded tasks. In this study, we aim at enhancing optic nerve head images obtained by Optical Coherence Tomography (OCT). However, instead of directly applying noise reduction techniques, we use multiple state-of-the-art image Super-Resolution (SR) methods. In SR, the low-resolution (LR) image is upsampled to match the size of the high-resolution (HR) image. With respect to image enhancement, the upsampled LR image can be considered as low quality, noisy image, and the HR image would be the desired enhanced version of it. We experimented with several image SR architectures, such as super-resolution Convolutional Neural Network (SRCNN), very deep Convolutional Network (VDSR), deeply recursive Convolutional Network (DRCN), and enhanced super-resolution Generative Adversarial Network (ESRGAN). Quantitatively, in terms of peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM), the SRCNN, VDSR, and DRCN significantly improved the test images. Although the ERSGAN showed the worst PSNR and SSIM, qualitatively, it was the best one.

Keywords: Medical image processing · OCT image enhancement · Image super resolution.

1 Introduction

In recent years, Deep Neural Networks (DNN) have shown great success in image processing and analysis, outperforming humans in some tasks such as image classification [20]. It has been a matter of time, when DNNs would find their way in the area of medical image processing. The enhancement of medical images is a task of high practical value since many of the current MRI or CT images are of low quality. Classical image enhancement methods are mostly based on histogram equalization techniques [19] which don't work well with medical images. Lately, there have been some studies where the DNN are used for image enhancement [15] and MRI scans denoising [8].

In this work, we focus on enhancing or rather denoising images obtained by Optical Coherence Tomography (OCT) [21]. The OCT technology has become

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50426-7_37

a widely used tool for assessing optic nerve head tissues and monitoring many ocular pathologies. However, the quality of OCT scans is hampered by mainly speckle noise [7] as well as some other artifacts [1]. There exist some methods, both hardware and software based, to denoise OCT scans. For example, the multi-frame averaging [10] is a hardware technique which greatly improves the image quality, but requires long scanning time. This inflicts discomfort and strain in many patients. Software based image denoising approaches include filtering [16] or some numerical methods [6].

So far, with respect to the OCT image processing, the usage of deep learning has been limited to image segmentation [22] and classification [14]. The only other work on OCT denoising we are aware of is [4].

The goal of the OCT image enhancement task is to improve the quality of a single OCT scan to match the quality of multi-frame averaged image produced by the OCT device. This would greatly reduce the time needed to obtain high-quality image, because one multi-frame scan can takes about 3 min while a single scan - only few seconds. From machine learning point of view, this is a supervised multiple regression task as depicted in Fig. 1, where the input is the low quality (LQ) single scan and the output is an enhanced high quality (HQ) image resembling the multi-frame OCT scan.

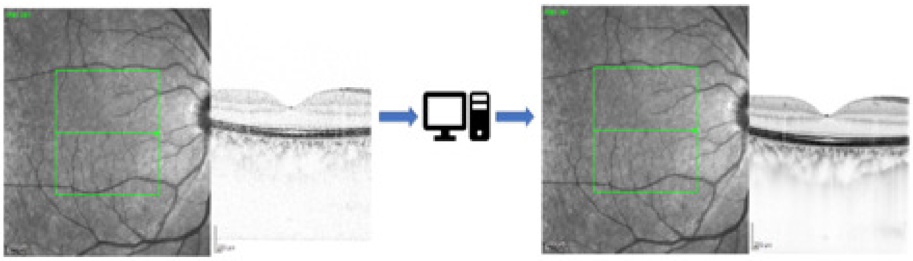


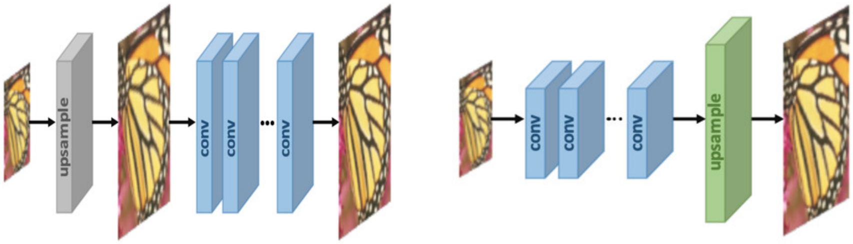
Fig. 1. The task of OCT scan enhancement. Low quality single scans are processed to obtain high quality images resembling the multi-frame scans as closely as possible.

In [4], researchers try to solve this task by adding Gaussian noise to the HQ multi-frame scans and use them as input to their denoising network based on the popular U-net [17]. This approach avoids problems with the image registration, because often there is a misalignment between single scans and their multi-frame counterparts. However, it ignores the actual speckle noise distribution which could be far from Gaussian and is OCT device dependent as well. Our approach differs in two main ways. First, we don't add artificial noise to the HQ multi-frame scans, but use the original LQ single scans. This apparently requires image registration which we performed using the excellent SimpleITK toolkit [2]. Second, we don't use DNN architectures targeted at image denoising, but adapt several state-of-the-art single images super resolution (SR) networks for the purposes of our task. They include super-resolution Convolutional Neural Network

(SRCNN), very deep Convolutional Network (VDSR), deeply recursive Convolutional Network (DRCN), and enhanced super-resolution Generative Adversarial Network (ESRGAN). The way we use the SR networks for image enhancement and some details for each of them are given in the next section. Later, we describe our data, experimental conditions and results we obtained.

2 Single image super resolution

Single image super resolution (SR) is a classical problem in computer vision where the aim is to recover high-resolution (HR) image from a single low-resolution (LR) image. With the rise of deep convolutional networks, the number of proposed solutions and network architectures has increased dramatically [24, 26]. In practice, since the HR image size is bigger, during processing, the input LR image has to be upsampled to match the size. There are different strategies where and how to do this in the processing pipeline. Two widely used approaches are shown in Fig. 2. In the first one, the LR image is upsampled in advance using some form of interpolation and then is passed to the SR model as in Fig. 2(a). The other way is to keep the LR image size and perform upsampling at the last processing step as in Fig. 2(b).

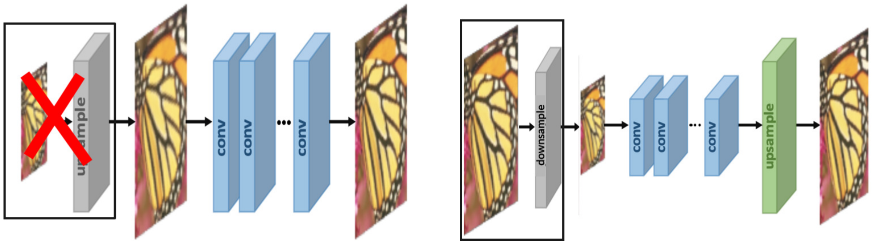


(a) Pre-upscaling SR (b) Post-upscaling SR

Fig. 2. Two widely used SR architectures where image upsampling is done either before a) or after b) the processing.

Since in our task, the size of the image should not change, we cannot use those SR architectures directly. However, if we remove the upsampling step in the case of Fig. 2(a), we end up with a system that essentially enhances the input image without changing its size. This is illustrated in Fig. 3(a). Unfortunately, this approach does not work with the architecture of Fig. 2(b). In this case, the upsampling step is part of the processing pipeline and its parameters are trainable. We solve this problem by first downsampling the input image and then passing it to the system as shown in Fig. 3(b).

In the next four subsections we describe briefly each of the SR networks we experimented with.



(a) In pre-upscaling SR, the first upsampling block is deleted

(b) In post-upscaling SR, a new downsampling block is added

Fig. 3. Changes made to accommodate the two SR architectures for image enhancement purposes.

2.1 Super resolution convolutional neural network (SRCNN)

The SRCNN [5] is a simple network consisting of two hidden convolutional layers as can be seen in Fig. 4. The input is supposed to be the upscaled version of the LR image, so the architecture corresponds to the pre-upsampling SR from Fig. 2(a).

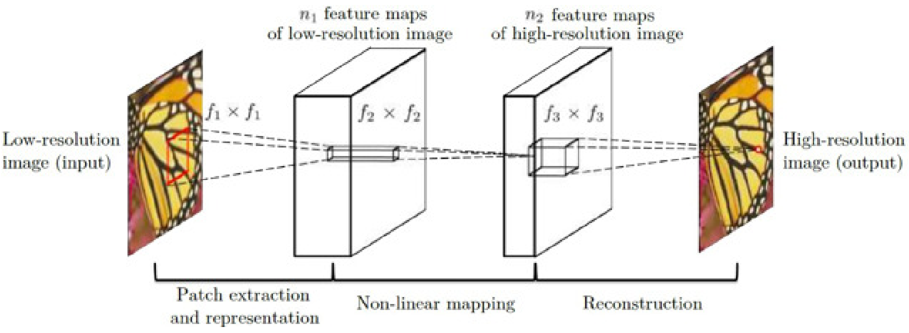

n_1 feature maps of low-resolution image
n_2 feature maps of high-resolution image
Low-resolution image (input)
$f_1 \times f_1$
$f_2 \times f_2$
$f_3 \times f_3$
High-resolution image (output)
Patch extraction and representation
Non-linear mapping
Reconstruction

Fig. 4. SRCNN architecture.

Each hidden layer performs standard convolutional operation with output clipped to be positive. The loss function is the mean squared error (MSE) between the output image $\tilde{Y}_i$ and the target HR image Y_i averaged over the training set:

$$L(\Theta) = \frac{1}{n} \sum_{i=1}^{n} \parallel \mathbf{Y}_i - \hat{\mathbf{Y}}_i \parallel^2 \tag{1}$$

The MSE loss function favors a high peak signal-to-noise ratio (PSNR) which is a widely-used metric for quantitative evaluation of SR quality. However, the PSNR is only partially related to the perceptual quality and in practice, sometimes images with high PSNR don't look perceptually very good.

2.2 Very deep convolutional network (VDSR)

Based on the popular VGG network [18] for image classification, the VDSR [11] consists of many convolutional layers with ReLU activation. The residual connection between the input and the last hidden layer (the long line in Fig. 5), forces the network to learn only the difference between the input and the target and as a result allows network to be much more deeper without vanishing/exploding gradients problem.

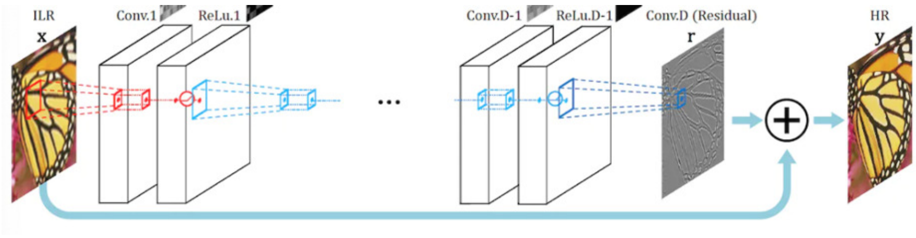


Fig. 5. VDSR architecture.

The input is an upsampled interpolated low-resolution (ILR) image, so the VDSR architecture falls into the pre-upsampling SR category as in Fig. 2(a). The loss function is computed as the Euclidean distance between the reconstructed image and the HR target image similar to Eq. (1). Therefore, the VDSR as the SRCNN favors high PSNR, but not high perceptual quality.

2.3 Deeply recursive convolutional network ((DRCN)

The VDSR [12] makes use of the same convolutional block up to 16 times. The main difference from the other structures is that a multi-supervised strategy is applied, so that the outputs of all the blocks are combined together as shown in Fig. 6. This approach not only allows gradients to flow easily through the network, but also encourages all the intermediate representations to reconstruct the HR image. In such multi-supervised approach, there are multiple objectives to minimize. The loss for the intermediate outputs is defined as:

$$l_1(\theta) = \frac{1}{2DN} \sum_{d=1}^{D} \sum_{i=1}^{N} \parallel \mathbf{y}_i - \hat{\mathbf{y}}_i^d \parallel^2 \tag{2}$$

where D is the number of recursions. For the final output with is a weighted sum of all intermediate outputs the loss is:

$$l_2(\theta) = \frac{1}{2N} \sum_{i=1}^{N} \parallel \mathbf{y}_i - \sum_{d=1}^{D} w_d \hat{\mathbf{y}}_i^d \parallel^2 \tag{3}$$

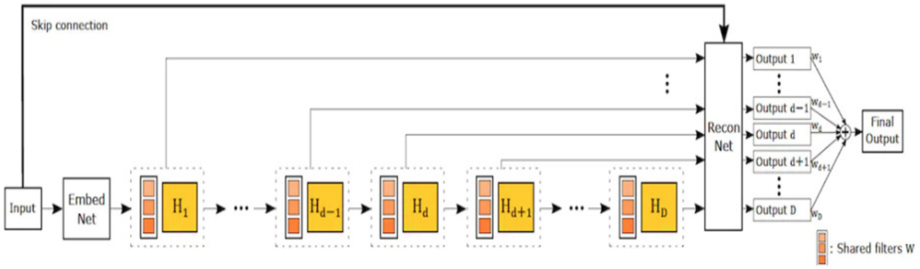


Fig. 6. DRCN architecture.

The final loss function includes both the l_1 and l_2 as well as a regularization term:

$$L(\theta) = \alpha l_1(\theta) + (1 - \alpha) l_2(\theta) + \beta \parallel \theta \parallel^2 \tag{4}$$

where α controls the trade-off between the intermediate and final losses and β - the amount of regularization. Note that all losses use the MSE criterion, so the DRCN also favors high PSNR images.

2.4 Enhanced super resolution generative adversarial network (ESRGAN)

The ESRGAN [23] is an improved version of the super resolution generative adversarial network (SRGAN) [13]. It consists of two networks - Generator and Discriminator working together. The structure of each of them is shown in Fig. 7. The Generator includes multiple blocks called residual in residual dense block (RRDB) which combine multi-level residual network and dense connections. The

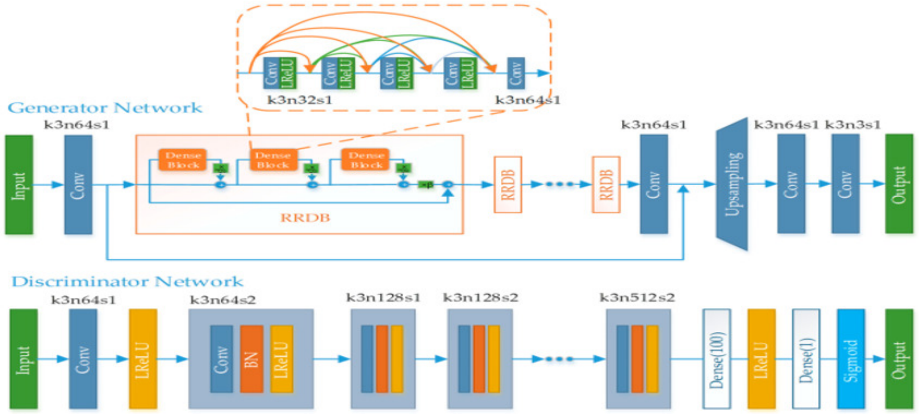


Fig. 7. ESRGAN architecture.

DOI: 10.1007/978-3-030-50426-7_37

upsampling block is located at the end of the pipeline, so the ESRGAN architecture is of the type shown in Fig. 2(b). The Discriminator has a simpler structure consisting of multiple convolution layers each followed by a batch normalization and Leaky ReLU activation. One important difference between the ESRGAN and other SR networks described above is that the Generator utilizes an improved version of the so called perceptual loss [9]. Originally, it is defined on the activation layers of a pre-trained network where the distance between two activated features is minimized. Thus, the Generator total loss is expressed as:

$$L_G^{tot} = L_{percep} + \lambda L_G + \eta L_1 \tag{5}$$

where $L_1 = \mathbb{E}_x \parallel G(x) - y \parallel_1$ is the 1-norm difference between the Generator output $G(x)$ given input image x and the target HR image y. Using such loss makes the ESRGAN to produce images of higher perceptual quality than the PSNR oriented networks.

3 Performance evaluation

There exist various quantitative performance metrics adopted in image processing among which the peak noise-to-signal ratio (PSNR) and the structural similarity index measure (SSIM) [25] are the most widely used. In [4], authors used pure SNR and SSID metrics, while we utilize the PSNR and SSID.

The MSE and PSNR between ground truth image I and reconstructed image $\hat{I}$ both of which have N pixels as defined as:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(I(i) - \hat{I}(i))^2 \tag{6}$$

$$PSNR = 10\log(\frac{L^2}{MSE}) \tag{7}$$

where $L = 255$ for 8-bit pixel encoding. Typical PSNR values vary from 20 to 40, higher is better.

On the other hand, the SSID is defined as:

$$SSIM(I, \hat{I}) = \frac{(2\mu_I\mu_{\hat{I}} + C_1)(2\sigma_{I\hat{I}} + C_2)}{(\mu_I^2 + \mu_{\hat{I}}^2 + C_1)(\sigma_I^2 + \sigma_{\hat{I}}^2 + C_2)} \tag{8}$$

where $C_1 = (k_1L)^2, C_2 = (k_2L)^2$ are constants for avoiding instability, $k_1 \ll 1, k_2 \ll 1$ are small constants, and μ and σ^2 are the mean and variance of the pixels intensity.

4 Experiments

4.1 Database

For the experiments, we used a small database of about 350 OCT scans. Some of the HQ multi-frame scans had several corresponding LQ single scans, so the

DOI: 10.1007/978-3-030-50426-7_37

same targets were used for those LQ images. Most of the HQ/LQ pairs required alignment and for this purpose we used the SimpleITK image registration toolkit [2]. Six HQ/LQ pairs were selected for testing, and the remaining data were split into training and validation sets by 9:1 ratio.

Since the number of scans is quite small, we did exhaustive data augmentation which includes horizontal and vertical flips, rotation by several different degrees, etc., commonly used in image processing practice. In addition, each scan was cropped into non-overlapping sub-images of size 224x224. Thus, we managed to increase the number of training data roughly 100 fold.

4.2 Results

Here, we present the results in terms of PSNR and SSIM metrics for each of the network architectures described in Section 2. In each case, we tried to tune the network hyper-parameters to achieve the best possible result. The results shown in the tables below reflect the performance dependence on the two most impactful parameters we found for each network.

All the networks were trained with up to 100 epochs and for testing we used the model obtained from the epoch where the PSNR of the validation data was the highest.

SRCNN results. The SRCNN is trained using small patches of size 33x33 taken from the input image with stride 14. This network in known to take many training iterations to achieve good performance, so we chose a small learning rate of 5.0e-6. We found that the batch size and the size of the filter of the first convolutional layer have the biggest influence on the SRCNN performance. The obtained PSNR and SSIM values are given in Table 1.

Table 1. SRCNN performance in terms of PSNR (dB) and SSIM.

Metric	Batch size	f_1 size		
		7x7	9x9	11x11
PSNR	64	**25.23**	25.15	24.81
	128	25.18	24.93	23.76
	256	24.96	25.10	24.38
SSIM	64	0.794	**0.798**	0.797
	128	0.795	0.798	0.790
	256	0.791	0.796	0.792

VDSR results. The patch size during the VDSR training was set to 41x41 with no overlap. We experimented with the number of convolutional blocks and the batch size. The learning rate was set to 0.001 and the other hyper-parameters were used as recommended by the VDSR developers. Table 2 shows the PSNR and SSID values obtained during the experiment.

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50426-7_37

Table 2. VDSR performance in terms of PSNR (dB) and SSIM.

Metric	Batch size	Number of blocks		
		8	16	32
PSNR	32	25.10	25.22	24.18
	64	25.12	25.38	25.30
	128	24.18	25.45	**25.51**
SSIM	32	0.791	0.785	0.543
	64	0.791	0.789	0.779
	128	0.791	**0.795**	0.778

DRCN results. With the DRCN, we used the same patch size as for the VDSR, but with stride 21 [11]. Initially, the learning rate was set to 0.01 and during training was decreased 10 times every time validation performance plateaus. The main architectural hyper-parameters of the DRCN are the number of blocks and the number of filters in each block. We varied those parameters and the results with batch size of 128 are presented in Table 3.

Table 3. DRCN performance in terms of PSNR (dB) and SSIM.

Metric	Filter number	Number of blocks		
		4	8	16
PSNR	16	24.81	25.02	25.12
	32	24.48	24.26	23.02
	48	18.07	22.37	**25.77**
SSIM	16	0.768	0.774	**0.778**
	32	0.761	0.762	0.535
	48	0.723	0.535	0.687

We have to note that we could not find a good trade-off between the intermediate loss l_1 and final loss l_2 functions given in Eq. (2) and Eq. (3) respectively. The best results we obtained when the combination parameter λ from Eq. (4) was set to 0.

ESRGAN results. In terms of parameters, this is the biggest network among all the networks we experimented with, and so is the number of possible hyper-parameters. Structurally, for the generator, important are the RDDB number, the RDB number in each RDDB as well as the number of convolutional layers and the number of filters. The discriminator's structure has no big influence on the performance. As can be seen from Table 4, in our case, the RDDB number

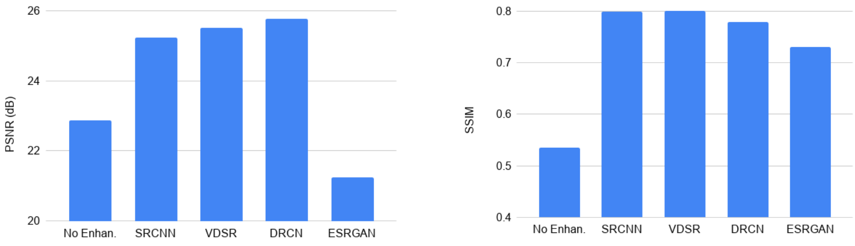
ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50426-7_37

and the filter number were the most sensitive to the ESRGAN performance. We couldn't obtain results for the case of RDDB number = 7 and filter number = 16 since the model was so big and did not fit in our GPU memory. The other parameters were as follows: number of RDBs inside each RRDB = 6, number of convolutional layers inside a RDB = 4, learning rate = 4.0e-4 with decay factor of 2. For training and evaluation of the ESRGAN, we used the ISR toolkit [3] and all the other parameters we left at their default values.

Table 4. ESRGAN performance in terms of PSNR (dB) and SSIM.

Metric	RDDB number	RDB filter number		
		4	8	16
PSNR	3	19.56	19.01	18.98
	5	19.53	**21.25**	18.92
	7	19.64	18.69	NA
SSIM	3	0.670	0.639	0.725
	5	0.432	0.722	**0.730**
	7	0.658	0.377	NA

Networks comparison. Here, we compare the best obtained performance from all the networks we evaluated in terms of PSNR and SSIM. Fig. 8 shows bar plots for each metric together with the case when no enhancement is applied. In terms of PSNR, the DRCN achieved the best result, while the best SSIM was achieved by the SRCNN and VDSR. In both cases, the obtained metrics values are much better than the baseline, i.e. the case of unprocessed single scan images.



(a) Best PSNR in dB.

(b) Best SSIM.

Fig. 8. Comparison of the networks best performances in terms of PSNR and SSIM with the baseline ("No Enhan.")

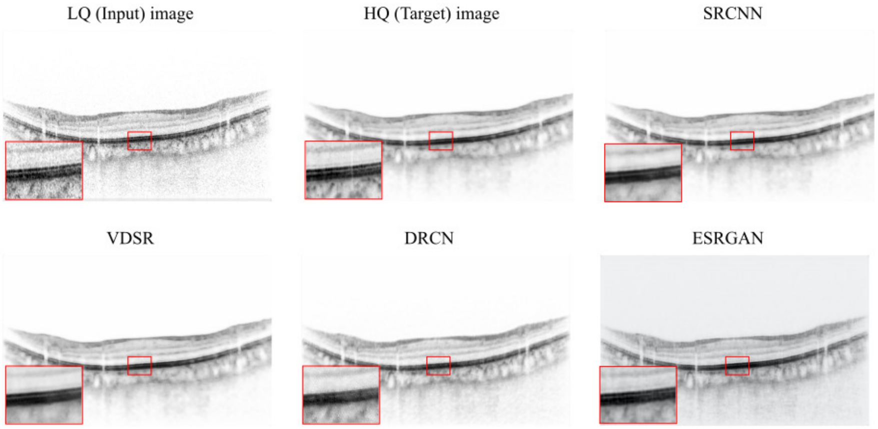


Fig. 9. Example test single scan (first row, left), the corresponding multi-frame averages scan (first row, center), and the results from each network.

The ERSGAN, however, showed PSNR even lower than the baseline. This can be explained with the fact that the ESRGAN is trained to improve the perceptual loss more than the mean absolute error (MAE) which is the L_1 in Eq. (5) and is related to the PSNR. To verify this hypothesis, we looked at all the test images enhanced by each of the networks and visually compared them. Indeed, the ESRGAN has produced the best looking images with sharper edges and higher contrast. As an example, we show one of the test single scans and its corresponding multi-frame scan as well as its enhanced versions by all the networks in Fig. 9.

5 Conclusion

In this study, we focused on enhancing single scans obtained from Optical Coherence tomography. They all contain speckle noise as well as some other artifacts making the interpretation of the OCT data cumbersome. Many OCT devices apply multi-frame averaging techniques to alleviate this problem, but this approach requires a lot of time and causes great discomfort to the patients.

Instead of using enhancing/denoising methods directly, we adopted some of the state-of-the-art deep neural networks designed for image super resolution. Since in many cases the low resolution images are first upscaled, an operation that degrades their quality, the SR networks essentially enhance those upscaled low resolution images.

We experimented with several SR networks such as SRCNN, VDSR, DRCN and ERSGAN and evaluated them quantitatively using PSNR and SSIM metrics.

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50426-7_37

Since all the networks but ESRGAN use MSE based loss function, they all achieved high PSNR values. However, qualitatively, the ESRGAN produced the best looking images which we attribute to the use of a perceptual loss function.

Our results are still preliminary, because the amount of training data was clearly insufficient to reliably train big networks such as DRCN or ESRGAN. Also, the OCT scans come from healthy patients only and many pathological artifacts haven not been learned. In addition, we expect scans from different OCT devices to have different noise distributions. All these problems we intend to address in the future.

Acknowledgment

We are grateful to Prof.Sekiryu from Fukushima Medical University for providing the OCT scan data.

References

1. Asrani, S., Essaid, L., Alder, B.D., Santiago-Turla, C.: Artifacts in spectral-domain optical coherence tomography measurements in glaucoma. JAMA ophthalmology **132**(4), 396–402 (2014)
2. Beare, R., Lowekamp, B., Yaniv, Z.: Image segmentation, registration and characterization in r with simpleitk. Journal of statistical software **86** (2018)
3. Cardinale, F., John, Z., Tran, D.: ISR. https://github.com/idealo/image-super-resolution (2018)
4. Devalla, S.K., Subramanian, G., Pham, T.H., Wang, X., Perera, S., Tun, T.A., Aung, T., Schmetterer, L., Thiéry, A.H., Girard, M.J.: A deep learning approach to denoise optical coherence tomography images of the optic nerve head. Scientific reports **9**(1), 1–13 (2019)
5. Dong, C., Loy, C.C., He, K., Tang, X.: Image super-resolution using deep convolutional networks. IEEE transactions on pattern analysis and machine intelligence **38**(2), 295–307 (2016)
6. Du, Y., Liu, G., Feng, G., Chen, Z.: Speckle reduction in optical coherence tomography images based on wave atoms. Journal of biomedical optics **19**(5), 056009 (2014)
7. Esmaeili, M., Dehnavi, A.M., Rabbani, H., Hajizadeh, F.: Speckle noise reduction in optical coherence tomography using two-dimensional curvelet-based dictionary learning. Journal of medical signals and sensors **7**(2), 86 (2017)
8. Jiang, D., Dou, W., Vosters, L., Xu, X., Sun, Y., Tan, T.: Denoising of 3d magnetic resonance images with multi-channel residual learning of convolutional neural network. Japanese journal of radiology **36**(9), 566–574 (2018)
9. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: European conference on computer vision. pp. 694–711. Springer (2016)
10. Kennedy, B.F., Hillman, T.R., Curatolo, A., Sampson, D.D.: Speckle reduction in optical coherence tomography by strain compounding. Optics letters **35**(14), 2445–2447 (2010)

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50426-7_37

11. Kim, J., Kwon Lee, J., Mu Lee, K.: Accurate image super-resolution using very deep convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1646–1654 (2016)
12. Kim, J., Kwon Lee, J., Mu Lee, K.: Deeply-recursive convolutional network for image super-resolution. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1637–1645 (2016)
13. Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., et al.: Photo-realistic single image super-resolution using a generative adversarial network. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4681–4690 (2017)
14. Lee, C.S., Baughman, D.M., Lee, A.Y.: Deep learning is effective for classifying normal versus age-related macular degeneration oct images. Ophthalmology Retina **1**(4), 322–327 (2017)
15. Lu, L., Zheng, Y., Carneiro, G., Yang, L.: Deep learning and convolutional neural networks for medical image computing. Advances in Computer Vision and Pattern Recognition; Springer: New York, NY, USA (2017)
16. Ozcan, A., Bilenca, A., Desjardins, A.E., Bouma, B.E., Tearney, G.J.: Speckle reduction in optical coherence tomography images using digital filtering. JOSA A **24**(7), 1901–1910 (2007)
17. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention. pp. 234–241. Springer (2015)
18. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
19. Suganya, P., Gayathri, S., Mohanapriya, N.: Survey on image enhancement techniques. International Journal of Computer Applications Technology and Research **2**(5), 623–627 (2013)
20. Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.A.: Inception-v4, inception-resnet and the impact of residual connections on learning. In: Thirty-First AAAI Conference on Artificial Intelligence (2017)
21. van Velthoven, M.E., Faber, D.J., Verbraak, F.D., van Leeuwen, T.G., de Smet, M.D.: Recent developments in optical coherence tomography for imaging the retina. Progress in retinal and eye research **26**(1), 57–77 (2007)
22. Venhuizen, F.G., van Ginneken, B., Liefers, B., van Grinsven, M.J., Fauser, S., Hoyng, C., Theelen, T., Sánchez, C.I.: Robust total retina thickness segmentation in optical coherence tomography images using convolutional neural networks. Biomedical optics express **8**(7), 3292–3316 (2017)
23. Wang, X., Yu, K., Wu, S., Gu, J., Liu, Y., Dong, C., Qiao, Y., Change Loy, C.: Esrgan: Enhanced super-resolution generative adversarial networks. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 0–0 (2018)
24. Wang, Z., Chen, J., Hoi, S.C.: Deep learning for image super-resolution: A survey. arXiv preprint arXiv:1902.06068 (2019)
25. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P., et al.: Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing **13**(4), 600–612 (2004)
26. Yang, W., Zhang, X., Tian, Y., Wang, W., Xue, J.H., Liao, Q.: Deep learning for single image super-resolution: A brief review. IEEE Transactions on Multimedia (2019)