Deep OCT image compression with convolutional neural networks

PENGFEI GUO,1,4 DAWEI LI,2,4 AND XINGDE LI2,3,*

1Department of Computer Science, Johns Hopkins University, Baltimore, MD 21218, USA
2Department of Biomedical Engineering, Johns Hopkins University, Baltimore, MD 21205, USA
3Department of Electrical and Computer Engineering, Johns Hopkins University, Baltimore, MD 21218, USA
4Equal contribution
*xingde@jhu.edu

Abstract: We report an end-to-end image compression framework for retina optical coherence tomography (OCT) images based on convolutional neural networks (CNNs), which achieved an image size compression ratio as high as 80. Our compression scheme consists of three parts: data preprocessing, compression CNNs, and reconstruction CNNs. The preprocessing module was designed to reduce OCT speckle noise and segment out the region of interest. Skip connections with quantization were developed and added between the compression CNNs and the reconstruction CNNs to reserve the fine-structure information. Two networks were trained together by taking the semantic segmented images from the preprocessing module as input. To train the two networks sensitive to both low and high frequency information, we leveraged an objective function with two components: an adversarial discriminator to judge the high frequency information and a differentiable multi-scale structural similarity (MS-SSIM) penalty to evaluate the low frequency information. The proposed framework was trained and evaluated on ophthalmic OCT images with pathological information. The evaluation showed reconstructed images can still achieve above 99% similarity in terms of MS-SSIM when the compression ratio reached 40. Furthermore, the reconstructed images after 80-fold compression with the proposed framework even presented comparable quality with those of a compression ratio 20 from state-of-the-art methods. The test results showed that the proposed framework outperformed other methods in terms of both MS-SSIM and visualization, which was more obvious at higher compression ratios. Compression and reconstruction were fast and took only about 0.015 seconds per image. The results suggested a promising potential of deep neural networks on customized medical image compression, particularly valuable for effective image storage and tele-transfer.

© 2020 Optical Society of America under the terms of the OSA Open Access Publishing Agreement

1. Introduction

There is an increasing demand for methods that can efficiently store and transfer imaging data when considering, in medicine alone, the mounting number of medical images collected at each hospital and the increasing image size (partially owing to improved resolution). Efficient imaging data compression, transfer and restoration are becoming particularly more critical for remote diagnosis and monitoring of diseases, such as age-related macular degeneration (AMD) detection with optical coherence tomography (OCT) [1,2]. For image compression, it is necessary to preserve the fine structural information (e.g., the detailed retinal layers information for ophthalmic OCT images) from the noisy background [3]. With recent advances in deep learning, convolutional neural networks (CNNs)-based framework offers a potential solution, capable of efficient compression and high-fidelity restoration.

While volumetric image compression is of great interest for both industrial standards (e.g., MPEG [4]) and clinical applications (e.g., 3-D OCT image compression [5]), 2-D image compression is also widely used in daily images and clinical data. In this study, we mainly
focus on 2-D compression of OCT images. The commonly used compression formats, such as BPG [6], WebP [7], JPEG [8], and JPEG2000 [9], become suboptimal when the compression ratio increases [3]. Customized compression methods, such as compressive sensing (CS) [10], provide an alternative; however, the CS-based methods, often involving convex optimization, are usually computationally expensive for achieving a high reconstruction accuracy [11–13]. Currently, CNNs provide an exciting new avenue for image compression [14–17]; yet most reported CNNs-based methods exhibit difficulty in preserving the fine structural information, especially at a high compression ratio.

In this paper, we report an efficient image compression and restoration framework based on CNNs to achieve a high compression ratio (up to 80:1) and restore the images with high fidelity (with a similarity to the original images better than 98%). The proposed scheme works in two stages: first, distinguishing diagnostic features of interest from noisy background by semantic segmentation [18–20]; second, compressing the segmented images by compression CNNs, interpreting the compressed files and restoring the images by reconstruction CNNs. This paper will present the detailed network structure and training detail of the CNNs for compression and reconstruction. The performance of the proposed framework will be demonstrated by quantitative and qualitative comparing with the commonly used compression methods including BPG, WebP, JPEG, JPEG2000, and a recurrent neural network (RNN)-based method [21]. This paper will end with a brief discussion on the novelty and prospect of the proposed framework.

2. Methods

Our compression framework consists of three modules: a data preprocessing module (Fig. 1(a)), a CNNs-based compression module and a CNNs-based reconstruction module (Fig. 1(b)). The data preprocessing module reduces OCT image noise and segments out the regions to be compressed, which helps efficiently train both the compression and reconstruction CNNs. The compression module generates the compressed file which consists of a bitstream that contains image information. The reconstruction module serves as a dictionary to interpret the bitstream to a high-resolution image. To reserve both high and low frequency information in the original images, we train the compression and reconstruction CNNs together with an adversarial objective function combined by a patch discriminator module and a differentiable multi-scale structural similarity (MS-SSIM) penalty module. The training, validation and testing data were obtained from an open source dataset provided by the Vision and Image Processing (VIP) Laboratory, Duke University [1].

2.1. Data preprocessing

2.1.1. Denoising and segmentation

OCT imaging is based on low-coherence interferometry and intrinsically suffers from speckle noise. Speckle noise not only degrades OCT image quality and makes it difficult to identify fine structural details, but also leads to fake guidance when we train the compression and reconstruction CNNs. To efficiently compress the OCT images and keep high-fidelity structural information, we chose to reduce the speckle noise in the OCT images as the first step (see Fig. 1(a)) by using the well-established denoising algorithm – Block-matching and 3D filtering (BM3D) which demonstrated superb performance in terms of peak signal-to-noise ratio and subjective visual quality [22–24]. To further increase the potential compression ratio, we segmented out the regions of interest to facilitate both of compression and reconstruction CNNs to effectively learn the feature representations from the ROIs and fully utilize the limited size of bitstreams. A semantic segmentation approach was used in preprocessing, which is based on the U-net architecture [19,25,26]. The U-Net segmentation architecture consisted of feature channels connecting the associated latent maps of down-sampling phase with the corresponding ones in
Fig. 1. (a) Schematic of data preprocessing procedure. There were three layers resulting from segmentation: 1. Inner limiting membrane (ILM: green line); 2. Bruch’s membrane (BW: redline); 3. Lower boundary corresponding to BM (LBM: yellow line). (b) Schematic of compression workflow, which is a conditional GANs model that contains a generator (including compression CNNs and reconstruction CNNs), and a discriminator for adversarial training (training phrase only). An additional differentiable MS-SSIM loss module is also deployed (training phrase only). The compressed binary output is quantized feature representations from the compression CNNs and serves as the input for the reconstruction CNNs.

the up-sampling phase, which enabled transmission of pixel-level localization information to up-sampling phase and vice versa during backpropagation [27–29]. The performance of the U-net architecture has demonstrated in various tasks in medical image analysis, especially, in biomedical image segmentation [30,31].

To run the image compression framework efficiently and avoid potential memory overflow, each original image with a size of 496 × 768 pixels was cropped out along the upper and lower boundaries [see the green and yellow lines in Fig. 1(a)] and was zero padded to the axial direction to make a final image of 256 × 512 pixels before compression [20]. Each rectangular image was then evenly divided into two square images with a size of 256 × 256 pixels each [see Fig. 1(a)]. All the images in the dataset were processed by the above procedures and divided into three independent groups: 740 images for training, 60 for validation, and 180 for testing. The images in each group were randomly selected without any overlap. The images in the training dataset were augmented by random cropping with scaling, random rotation and random horizontal flipping in each training epoch to prevent potential overfitting [19]. The compression CNNs and reconstruction CNNs were trained simultaneously. In the validation phase, hyper-parameters were fine-tuned, such as the initial learning rate, the number of skip connections, the quantization
levels, and the number of channels of each convolution layer etc. We tested the as-developed compression and reconstruction CNNs with the independent 180 images and made the final quality assessment based on multi-scale structural similarity (MS-SSIM). 5-fold cross-validation was adapted to report the performance.

2.2. Architecture of the networks

2.2.1. Compression CNNs and reconstruction CNNs

The schematic of proposed compression and reconstruction CNNs with a quantizer is shown in Fig. 2. The compression CNNs contracted the dimension of the input images layer by layer, and the reconstruction CNNs expanded the dimension in a symmetrical way. In detail, six layers in the compression CNNs learned the contextual features and hierarchically contracted the dimension of the input images. A quantizer residing between compression and reconstruction CNNs quantized the feature representations from three skip connections and the output of the last layer of compression CNNs to generate a bitstream (the compressed image). Three skip connections (the three red arrows from left to right in Fig. 2) added additional concatenations between two CNNs and fed the image information from different scales into the reconstruction CNNs via the quantizer. Skip connections preserved the fine structure information of the original images and transferred it to the corresponding reconstruction layers. The dimension of each input to the compression CNNs was 256 $\times$ 256 (H x W) pixels. The first layer applied convolution with 64 filters (each with a size of 7 $\times$ 7 pixels) and stride 1 to generate feature maps with the size of 256 $\times$ 256 $\times$ 64, followed by a normalization layer, and a Leaky Rectified Linear Unit (Leaky ReLu) activation to prepare the down-sampling phase. Five more layers with similar modules followed with a down sampling rate of 2 $\times$ 2. At the bottom of the compression CNNs, the most contracted feature maps with the dimension of ($H_32 \times W_32$) $\times$ $C_0$ (where $C_0$ is the number of channels) were passed through the quantizer to the reconstruction CNNs. The reconstruction CNNs symmetrically reversed the process in the compression CNNs but we replaced the Leaky ReLu activation functions by the Rectified Linear Unit (ReLu) activation functions to achieve better reconstruction quality [32]. The output images from the reconstruction CNNs had exactly the same dimension as the input images. In order to enhance the resolution of the up-sampling process, three skip connections concatenated a part of the output feature maps of the third, fourth, and fifth layer of the compression CNNs and the reconstruction CNNs through the quantizer [33–37]. In order to control the size of the bitstream (the compressed image), we used $C_0$, $C_1$, , and $C_3$ to control the number of channels of the feature maps from the output of the last layer of the compression CNNs and each skip connection, respectively. A specific compression ratio could be achieved by multiple combinations of $C_0$, $C_1$, $C_2$, and $C_3$. Based on experimental results, we found the empirical combinations that generated the best reconstruction quality and used those in final version of the framework.

2.2.2. Multi-scale quantization and compressed output

Between the compression and reconstruction CNNs, a quantizer quantized the feature maps from different compression layers and generated a bitstream as output (red and black arrows in Fig. 2). The output contained two parts: (1) the bitstream from the last compression layer, which was the major part of the residual information, and (2) 3 skip connections from the “upper” layers, which facilitated enhancing the fine structures in the reconstructed images [38].

In the quantization process, a scalar variant method was used to quantize the feature maps in L quantization levels and then encoded quantized feature maps to a bitstream [16,39]. For a given quantization level L, the compression ratio is defined as the ratio of the spatial dimension of the
Fig. 2. Schematic of proposed generator in a conditional GANs model, which contains compression CNNs (blue blocks on the left) and reconstruction CNNs (green blocks on the right). Leaky Rectified Linear Unit (Leaky ReLu) and Rectified Linear Unit (ReLu) induce nonlinearity for efficient training [7]. Convolution layers with stride 2 reduce feature maps by a factor of 2 along each dimension [13]. Arrows with different colors indicate different operations in the networks: the red arrows indicate quantized skip connections from the compression CNNs to the reconstruction CNNs via a quantizer; the black arrow represents the quantization operation on the most contracted feature map from the last layer of the compression CNNs; arrows with other colors represent the different combinations of convolution layers, normalization layers and activation functions. Please see the Section 2.2.1 for more details.

Compressed outputs to the input dimension, i.e.,

\[
\text{Compression Ratio (CR)} = \frac{H_{\text{input}} \cdot W_{\text{input}} \cdot S}{\sum_{i=1}^{n} H_i \cdot W_i \cdot C_i \cdot \log_2 L}. \tag{1}
\]

Here \(H_{\text{input}}, W_{\text{input}}\) is the height and width of input images, respectively; \(S\) is the bit depth of the input image; \(n\) is the number of connections passing through the quantizer; \((H_i \cdot W_i \cdot C_i)\) is the spatial dimension of the feature maps in each connection. In our work (Fig. 2), the dimensions of the input feature maps were \((8 \times 8 \times C_0)\), \((16 \times 16 \times C_1)\), \((32 \times 32 \times C_2)\), and \((64 \times 64 \times C_3)\), and the quantitation level \(L\) was 7. By changing the number of channels in the feature maps \((C_0, C_1, C_2, \text{and } C_3)\), we could achieve a compression ratio ranging from 10 to 80.

2.3. Training

2.3.1. Objective functions and optimization

In the training phase, the compression CNNs and reconstruction CNNs were trained together as the generator of conditional GANs [32]. The generator took original images as input and produced reconstructed images as output and the loss was represented by the difference between the original images and the reconstructed images in term of the proposed objective function. The training of the proposed framework was thus self-supervised. Our objective function consisted of two parts: a patched decoder as the discriminator of the conditional GANs to identify the fine-structure difference between the reconstructed images and the original images, and an MS-SSIM loss function to evaluate the difference between the input and output on a larger scale, which forced the generator to model the low-frequency information more efficiently [32]. The whole objective function \(L_G\) used in our framework is defined as below:

\[
L_G = \arg\min_G \max_D L_{\text{GAN}}(G, D) + \lambda L_{\text{MS–SSIM}}(G), \tag{2}
\]
Research Article

Vol. 11, No. 7 / 1 July 2020 / Biomedical Optics Express

3548

where $G$ consists of the compression CNNs and reconstruction CNNs, $D$ indicates the PatchGAN discriminator minimizing this objective function in an adversarial way, and $\lambda$ is the weight of the MS-SSIM loss. In our case, an empirical weight $\lambda = 100$ was selected. Specifically, the loss of the PatchGAN discriminator can be expressed as below [32]:

$$L_{\text{cGAN}}(G, D) = \mathbb{E}_{x,y} [\log D(x, y)] + \mathbb{E}_{x,z} [\log(1 - D(x, G(x, z)))] .$$  (3)

Here $x$ represents the original images; $y$ represents the reconstructed images; $z$ represents random noise, which is optional in our model; $E$ represents mathematical expectation. The task for the PatchGAN discriminator was to identify whether each $N \times N$ patch of the input image was real or not. Here, $N \times N$ was the size of the receptive field for the PatchGAN discriminator and the patch size was set to be $70 \times 70$. The discriminator consisted of four convolution layers. In each layer, the input feature maps were convoluted by $4 \times 4$ filters with a stride step of 2 and then, sequentially went through an instance normalization layer and a Leaky ReLu activation function. After the last layer, the output feature maps were mapped to a one-channel output followed by a sigmoid function activation to generate a final evaluation score [28,32,40]. For the optimization procedure, we used the Adam optimizer to minimize the loss function with an exponential decay rates $\beta = (0.9, 0.999)$ for the moment estimates [41]. Training hyperparameters were set as follows: constant learning rate of $2.0 \times 10^{-4}$ for the first 100 epochs then linearly decaying to 0; 200 maximum epochs; batch size of 1.

3. Results

In order to quantitatively assess the performance of our framework, we combined an MS-SSIM evaluator and direct human inspection that is often more sensitive to certain types of distortions than others [30,42]. The MS-SSIM evaluator produced a score between 0 and 1, with a higher value implying a closer match between the reconstructed and original images. The MS-SSIM evaluator is defined by below equation:

$$MS - SSIM(x, y) = l_M(x, y) \cdot \prod_{j=1}^{M} c_j(x, y) s_j(x, y) ,$$  (4)

where $x$ represents the original image and $y$ represents the reconstructed image. The algorithm iteratively applied a low-pass filter and down-sampled the filtered image by a factor of 2. We denote the original image as Scale 1, and the highest scale as Scale $M$. $l_j(x, y)$, $c_j(x, y)$, and $s_j(x, y)$ respectively represent the luminance comparison, contrast comparison, and structure comparison at the $j$-th scale [40,42].

3.1. Comparison with other methods

To fairly compare the compression performance of each method, all the test images used for the assessment metrics were preprocessed in the same fashion as mentioned in Section 2.1.1. We compared the performance of the proposed framework with other methods shown in Table 1 under a set of compression ratios (10, 20, 40, and 80). By controlling the quantization level $L$ and the channel number of the latent feature maps from the compression CNNs, we trained a set of models at each aforementioned compression ratio. We performed a 5-fold cross-validation for each method and the quantitative comparison results were reported in Table 1 with the results presented by the mean and the corresponding 95% confidence interval. Although the RNN-based approach achieved slightly better performance than other non-learnable methods as shown in Table 1, its reconstruction was significantly time-consuming due to the patch-based iterative reconstruction [1,21]. In this paper, we mainly focused on the discussion and comparison with BPG which outperforms other non-learnable methods. Both the proposed approach [Figs. 3(a)
and 3(b)] and BPG [Figs. 3(b) and 3(c)] achieved a high similarity index in terms of MS-SSIM at low compression ratios (CR < 20), but the proposed method was able to maintain a high similarity index and far outperform BPG at high compression ratios. To further evaluate the performance of the proposed method, we applied the previously trained networks to a second set of ophthalmic OCT images from an independent source [2], and the performance is reported in Table 2.

Table 1. Quantitative comparison of different methods in term of the average of MS-SSIM. The corresponding 95% confidence intervals are reported in the brackets. CR denotes the compression ratio. EDT denotes the encoding-decoding time per image in seconds.

| CR  | Ours | BPG [6] | WebP [7] | JPEG [8] | JPEG 2000 [9] | RNN [21] |
|-----|------|---------|----------|----------|---------------|-----------|
| 80  | 0.985| 0.973   | 0.956    | 0.922    | 0.955         | 0.979     |
|     | [0.984,0.986] | [0.972,0.974] | [0.955,0.957] | [0.921,0.923] | [0.954,0.956] | [0.975,0.984] |
| 40  | 0.989| 0.987   | 0.981    | 0.973    | 0.979         | 0.987     |
|     | [0.988,0.990] | [0.987,0.988] | [0.980,0.982] | [0.972,0.974] | [0.979,0.98] | [0.982,0.992] |
| 20  | 0.992| 0.992   | 0.990    | 0.989    | 0.989         | 0.993     |
|     | [0.991,0.993] | [0.992,0.993] | [0.990,0.991] | [0.988,0.989] | [0.988,0.990] | [0.990,0.996] |
| 10  | 0.994| 0.994   | 0.991    | 0.993    | 0.993         | 0.994     |
|     | [0.993,0.995] | [0.993,0.995] | [0.990,0.991] | [0.992,0.993] | [0.992,0.993] | [0.991,0.997] |
| EDT(s) | 1.51E-02 | 1.09E-01  | 2.06E-02 | 1.52E-02  | 2.03E-02     | 1.85E+01 |

Table 2. Quantitative comparison of different methods in term of the average of MS-SSIM on a second set of ophthalmic OCT images from an independent source [2]. The corresponding 95% confidence intervals are reported in the brackets.

| CR  | Ours | BPG [6] | WebP [7] | JPEG [8] | JPEG 2000 [9] | RNN [21] |
|-----|------|---------|----------|----------|---------------|-----------|
| 80  | 0.935| 0.927   | 0.929    | 0.882    | 0.893         | 0.843     |
|     | [0.926,0.943] | [0.918,0.936] | [0.922,0.936] | [0.873,0.890] | [0.883,0.904] | [0.823,0.864] |
| 40  | 0.958| 0.956   | 0.949    | 0.942    | 0.935         | 0.910     |
|     | [0.952,0.965] | [0.948,0.963] | [0.942,0.956] | [0.935,0.950] | [0.927,0.944] | [0.898,0.923] |
| 20  | 0.963| 0.976   | 0.971    | 0.972    | 0.964         | 0.961     |
|     | [0.958,0.968] | [0.969,0.984] | [0.964,0.979] | [0.965,0.979] | [0.956,0.971] | [0.956,0.967] |
| 10  | 0.968| 0.987   | 0.985    | 0.985    | 0.981         | 0.977     |
|     | [0.963,0.972] | [0.980,0.994] | [0.978,0.993] | [0.978,0.992] | [0.974,0.989] | [0.973,0.981] |

We notice that the performance of our proposed method degrades slightly at low compression ratios, which is mainly due to the additional noise associated with a different image acquisition system that our networks had not been trained for. We can observe similar degradation in all tested methods. However, at high compression ratios (CR of 40 or higher), our proposed method still outperforms others.

Comparing the two approaches, the proposed framework preserved fine structures, such as the Bruch’s membrane (BM), with an MS-SSIM value of 0.985 even at a high compression ratio of 80 [see the corresponding zoomed-in images in Fig. 3(b)]. In contrast, the performance of BPG severely deteriorated (with an MS-SSIM value of 0.973) at a compression ratio of 80 with blurred fine structures. In comparison, the reconstructed images from the proposed framework exhibited highly consistent visual appearance over a large compression ratio range as shown in Fig. 3. To further demonstrate the practicability of the proposed method, we also illustrate the performance of reconstruction for images with diseases [i.e., age-related macular degeneration (AMD) and diabetic macular edema (DME)] in Fig. 4. The first row shows an early case of AMD,
where the hyperreflective foci (HRF) and drusen appearing as retinal pigment epithelium (RPE) deformation were clearly preserved in the compressed image. The second and third rows show examples of DME images, where a giant outer nuclear layer (ONL) cyst in the inner nuclear layer and HRFs of different grades could be clearly identified in the compressed images.
4. Discussion

In this paper, we developed an OCT image compression framework which could preserve fine structural features at a high compression ratio (as high as 80) with deep neural networks. It outperformed the general compression formats in both similarity index and visual examination. The proposed framework demonstrated superb performance mainly due to three factors: (1) CNNs efficiently compress and reconstruct the image with high fidelity for both low and high frequency information by introducing an adversarial PatchGAN discriminator and an MS-SSIM penalty in the proposed objective function [43,44]; (2) spackle noise reduction and selection of only regions of interest (ROI) make the model focused on specific structural information [45]; (3) customized skip connections enhance the reconstruction quality by preserving fine structural information at different scales (from feature maps with a size of 1/4, 1/8 and 1/16 of the original images) [38].

Figure 5 shows the effectiveness of data preprocessing (i.e., denoising and segmentation) and skip connections in the proposed compression framework. We trained the models with the same hyper-parameters and reconstructed the images at the same compression ratio. Figure 5(a) illustrates the visual comparison among the reconstructed images from the proposed framework and those reconstructed either without image preprocessing (denoising and ROI segmentation), or without skip connections in the CNNs, or without both. Without image preprocessing, the
proposed framework was “fouled” by spackle noise and tried to reconstruct it. As a result, it degraded the compressed image. Without skip connections, it was more difficult to preserve the fine structure information and the compressed image was blurred. Without both preprocessing and skip connections, the compressed image was far more suboptimal. Figure 5(b) shows the quantitative comparison, where data preprocessing and skip connections clearly helped reduce image distortion in terms of the MS-SSIM index. For deploying the proposed method to other OCT images collected with different systems or for non-ophthalmic applications, it is suggested that the model be re-trained to achieve a desired performance. With a larger training dataset and upgraded network frameworks, it is feasible to push deep-learning based compression algorithms from cross-sectional images to volumetric ones, which will be our next direction for the future work.

Due to the three factors mentioned above, our model results in excellent performance comparable to the latest published general image compression methods based on CNNs, in which the networks had more complex architectures and were trained with more abundant dataset [17,21,46–48]. Furthermore, the computational efficiency of the proposed framework is also notable. It only took about 0.015 seconds to compress and reconstruct an image on a Ubuntu 18.04 computer with an NVIDIA 2080Ti GPU and PyTorch implementation.

**Funding**

National Institutes of Health (R01CA200399, R01HL121788).

**Acknowledgments**

This research was supported in part by the National Institutes of Health (R01HL121788, R01CA200399). The authors would also like to thank Professor Sina Farsiu and the Vision and Image Processing (VIP) Laboratory at Duke University for their open source OCT image set.
Disclosures

The authors declare that there are no conflicts of interest related to this article. The code will be available by emailing a request to jhu.bme.bit@gmail.com.

References

1. P. P. Srinivasan, L. A. Kim, P. S. Mettu, S. W. Cousins, G. M. Comer, J. A. Izatt, and S. Farsiu, “Fully automated detection of diabetic macular edema and dry age-related macular degeneration from optical coherence tomography images,” Biomed. Opt. Express 5(10), 3568–3577 (2014).
2. R. Rasti, H. Rabbani, A. Mehridehnavi, and F. Hajizadeh, “Macular OCT classification using a multi-scale convolutional neural network ensemble,” IEEE Trans. Med. Imaging 37(4), 1024–1034 (2018).
3. F. Liu, M. Hernandez-Cabronero, V. Sanchez, M. Marcellin, and A. Bilgin, “The current role of image compression standards in medical imaging,” Inform. 8(4), 131 (2017).
4. D. Le Gall, “MPEG: A video compression standard for multimedia applications,” Commun. ACM 34(4), 46–58 (1991).
5. L. Fang, S. Li, X. Kang, J. A. Izatt, and S. Farsiu, “3-D adaptive sparsity based image compression with applications to optical coherence tomography,” IEEE Trans. Med. Imaging 34(6), 1306–1320 (2015).
6. U. Albalawi, S. P. Mohanty, and E. Kougioumtzis, “A hardware architecture for better portable graphics (BPG) compression encoder,” in 2015 IEEE International Symposium on Nanoelectronic and Information Systems, (IEEE, 2015), 291–296.
7. L. Lian and W. Shilei, “Webp: A New Image Compression Format Based on VP8 Encoding,” Microcontrollers & Embedded Systems 3 (2012).
8. G. K. Wallace, “The JPEG still picture compression standard,” IEEE Trans. Consumer Electron. 38(1), xviii–xxxiv (1992).
9. D. Taubman and M. Marcellin, JPEG2000 Image Compression Fundamentals, Standards and Practice: Image Compression Fundamentals, Standards and Practice (Springer Science & Business Media, 2012), Vol. 642.
10. D. L. Donoho, “Compressed sensing,” IEEE Trans. Inform. Theory 52(4), 1289–1306 (2006).
11. A. M. Abdulghani and E. Rodriguez-Villegas, “Compressive sensing: From “Compressing while Sampling” to “Compressing and Secure while Sampling,” in 2010 Annual International Conference of the IEEE Engineering in Medicine and Biology, (2010), 1127–1130.
12. Y. C. Eldar and G. Kutyniok, Compressed Sensing: Theory and Applications (Cambridge University Press, 2012).
13. S. Stanković, I. Orović, and E. Sejdić, Multimedia Signals and Systems: Basic and Advanced Algorithms for Signal Processing (Springer, 2015).
14. P. Wang, V. M. Patel, and I. Hacihaliloglu, “Simultaneous segmentation and classification of bone surfaces from ultrasound using a multi-feature guided cnn,” in International Conference on Medical Image Computing and Computer-Assisted Intervention, (Springer, 2018), 134–142.
15. Y. Blau and T. Michaeli, “The perception-distortion tradeoff,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, (2018), 6228–6237.
16. E. Agustsson, F. Mentzer, M. Tschannen, L. Cavigelli, R. Timofte, L. Benini, and L. V. Gool, “Soft-to-hard vector quantization for end-to-end learning compressible representations,” in Advances in Neural Information Processing Systems, (2017), 1141–1151.
17. L. Theis, W. Shi, A. Cunningham, and F. Huszár, “Lossy image compression with compressive autoencoders,” arXiv preprint arXiv:1703.00395 (2017).
18. S. J. Chiu, M. J. Allingham, P. S. Mettu, S. W. Cousins, J. A. Izatt, and S. Farsiu, “Kernel regression based segmentation of optical coherence tomography images with diabetic macular edema,” Biomed. Opt. Express 6(4), 1172–1194 (2015).
19. A. G. Roy, S. Conjeti, S. P. K. Karri, D. Sheet, A. Katouzian, C. Wachinger, and N. Navab, “ReLayNet: retinal layer and fluid segmentation of macular optical coherence tomography using fully convolutional networks,” Biomed. Opt. Express 8(8), 3627–3642 (2017).
20. D. Liu, J. Wu, Y. He, X. Yao, W. Yuan, D. Chen, H.-C. Park, S. Yu, J. L. Prince, and X. D. Li, “Parallel deep neural networks for endoscopic OCT image segmentation,” Biomed. Opt. Express 10(3), 1126–1135 (2019).
21. G. Toderici, D. Vincent, N. Johnston, S. Jin Hwang, D. Minnen, J. Shor, and M. Covell, “Full resolution image compression with recurrent neural networks,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, (2017), 5306–5314.
22. M. Maggioni, V. Katkovnik, K. Egiazarian, and A. J. T. O. I. P. Foi, “Nonlocal transform-domain filter for volumetric data denoising and reconstruction,” IEEE Trans. on Image Process. 22(1), 119–133 (2013).
23. K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, “Image denoising by sparse 3-D transform-domain collaborative filtering,” IEEE Trans. on Image Process. 16(8), 2080–2095 (2007).
24. P. Wang, H. Zhang, and V. M. Patel, “SAR image despeckling using a convolutional neural network,” IEEE Signal Process. Lett. 24(12), 1763–1767 (2017).
25. V. Iglovikov and A. Shvets, “Ternausnet: U-net with vgg11 encoder pre-trained on imagenet for image segmentation,” arXiv preprint arXiv:1801.05746 (2018).
26. F. Milletari, N. Navab, and S.-A. Ahmadi, “V-net: Fully convolutional neural networks for volumetric medical image segmentation,” in 2016 Fourth International Conference on 3D Vision (3DV), (IEEE, 2016), 565–571.

27. S. Lefkimmiatis, A. Bourquard, and M. Unser, “Hessian-based norm regularization for image restoration with biomedical applications,” IEEE Trans. on Image Process. 21(3), 983–995 (2012).

28. A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagener classification with deep convolutional neural networks,” in Advances in neural information processing systems, 2012), 1097–1105.

29. L. Bottou, “Stochastic gradient descent tricks,” in Neural networks: Tricks of the trade (Springer, 2012), pp. 421–436.

30. J. Oliveira, S. Pereira, L. Gonçalves, M. Ferreira, and C. A. Silva, “Multi-surface segmentation of OCT images with AMD using sparse high order potentials,” Biomed. Opt. Express 8(1), 281–297 (2017).

31. L. Fang, D. Cunefare, C. Wang, R. H. Guymer, S. Li, and S. Farsiu, “Automatic segmentation of nine retinal layer boundaries in OCT images of non-exudative AMD patients using deep learning and graph search,” Biomed. Opt. Express 8(5), 2732–2744 (2017).

32. P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, “Image-to-image translation with conditional adversarial networks,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2017), 1125–1134.

33. J. Johnson, A. Alahi, and L. Fei-Fei, “Perceptual losses for real-time style transfer and super-resolution,” in European Conference on computer Vision, (Springer, 2016), 694–711.

34. Y. Han and J. C. Ye, “Framing U-Net via deep convolutional framelets: Application to sparse-view CT,” IEEE Trans. Med. Imaging 37(6), 1418–1429 (2018).

35. X. Wang and A. Gupta, “Generative image modeling using style and structure adversarial networks,” in European Conference on Computer Vision, (Springer, 2016), 318–335.

36. D. Voo, N. Kim, S. Park, A. S. Pack, and I. S. Kweon, “Pixel-level domain transfer,” in European Conference on Computer Vision, (Springer, 2016), 517–532.

37. Y. Zhou and T. L. Berg, “Learning temporal transformations from time-lapse videos,” in European conference on computer vision, (Springer, 2016), 262–277.

38. O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” in International Conference on Medical Image Computing and Computer-assisted Intervention, (Springer, 2015), 234–241.

39. F. Mentzer, E. Agustsson, M. Tschannen, R. Timofte, and L. Van Gool, “Conditional probability models for deep image compression,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018), 4394–4402.

40. Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: from error visibility to structural similarity,” IEEE Trans. on Image Process. 13(4), 600–612 (2004).

41. D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” arXiv preprint arXiv:1412.6980 (2014).

42. Z. Wang, E. P. Simoncelli, and A. C. Bovik, “Multiscale structural similarity for image quality assessment,” in The Thirty-Seventh Asilomar Conference on Signals, Systems & Computers, 2003, (IEEE, 2003), 1398–1402.

43. C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, and Z. Wang, “Photo-realistic single image super-resolution using a generative adversarial network,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2017), 4681–4690.

44. J. Pan, Y. Liu, J. Dong, J. Zhang, J. Ren, J. Tang, Y.-W. Tai, and M.-H. Yang, “Physics-based generative adversarial models for image restoration and beyond,” arXiv preprint arXiv:1808.00605 (2018).

45. S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” arXiv preprint arXiv:1502.03167 (2015).

46. O. Rippel and L. Bourdev, “Real-time adaptive image compression,” in Proceedings of the 34th International Conference on Machine Learning-Volume 70, (JMLR. org, 2017), 2922–2930.

47. N. Johnston, D. Vincent, D. Minnen, M. Covell, S. Singh, T. Chinen, S. Jin Hwang, J. Shor, and G. Toderici, “Improved lossy image compression with priming and spatially adaptive bit rates for recurrent networks,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018), 4385–4393.

48. J. Ballé, V. Laparra, and E. P. Simoncelli, “End-to-end optimized image compression,” arXiv preprint arXiv:1611.01704 (2016).