

Article

An Image Compression Method for Video Surveillance System in Underground Mines Based on Residual Networks and Discrete Wavelet Transform

Fan Zhang ^{1,2,*}, Zhichao Xu ¹ , Wei Chen ^{3,4} , Zizhe Zhang ³, Hao Zhong ¹, Jiaxing Luan ¹ and Chuang Li ¹

¹ School of Electrical and Information Engineering, China University of Mining and Technology (Beijing), Beijing 100083, China; generalxzc@hotmail.com (Z.X.); anatole_hao@163.com (H.Z.); sqt1800407113@student.cumtb.edu.cn (J.L.); zqt1800407138g@student.cumtb.edu.cn (C.L.)

² Institute of Intelligent Mining and Robotics, China University of Mining and Technology (Beijing), Beijing 100083, China

³ School of Computer Science and Technology and Mine Digitization Engineering Research Center of the Ministry of Education, China University of Mining and Technology, Xuzhou 221116, China; chenwdavior@163.com (W.C.); z32000@126.com (Z.Z.)

⁴ School of Earth and Space Sciences, Peking University, Beijing 100871, China

* Correspondence: zf@cumtb.edu.cn

Received: 28 November 2019; Accepted: 13 December 2019; Published: 17 December 2019



Abstract: Video surveillance systems play an important role in underground mines. Providing clear surveillance images is the fundamental basis for safe mining and disaster alarming. It is of significance to investigate image compression methods since the underground wireless channels only allow low transmission bandwidth. In this paper, we propose a new image compression method based on residual networks and discrete wavelet transform (DWT) to solve the image compression problem. The residual networks are used to compose the codec network. Further, we propose a novel loss function named discrete wavelet similarity (DW-SSIM) loss to train the network. Because the information of edges in the image is exposed through DWT coefficients, the proposed network can learn to preserve the edges better. Experiments show that the proposed method has an edge over the methods being compared in regards to the peak signal-to-noise ratio (PSNR) and structural similarity (SSIM), particularly at low compression ratios. Tests on noise-contaminated images also demonstrate the noise robustness of the proposed method. Our main contribution is that the proposed method is able to compress images at relatively low compression ratios while still preserving sharp edges, which suits the harsh wireless communication environment in underground mines.

Keywords: underground mines; intelligent surveillance; residual networks; compressed sensing; image compression; image restoration; discrete wavelet transform

1. Introduction

1.1. The Image Compression Demand from Underground Mines

Coal is one of the major resources in China. In the foreseeable future, China will still be the largest consumer and the producer of coal [1]. Therefore, it is of great importance to research into technologies that contribute to the advancement in intelligent mine monitoring and safe mining practices.

One of the key components of intelligent mine monitoring is the video surveillance system since visual information plays a key role in how a human perceives the world. Because digital images usually require large storage, it is natural to think of transmitting images with high bandwidth

channels, like cable networks. Although cable networks could potentially provide enough bandwidth, they are inflexible in that the cable networks are fixed and have to expand as the working surface expands. In favor of mobility, wireless networks are usually chosen as the information channel in mines. However, the bandwidth can be limited because of relatively limited narrow spaces, harsh environment diffraction, attenuation, and multi-path effect in underground mines. The problem can be especially serious when disasters such as explosion and collapse occur [2]. Therefore, it is necessary to investigate image compression methods in order to save the transmission bandwidth.

1.2. From Conventional Image Compressing to Compressed Sensing

There have been vast investigations into the field of image compression. Among the researches, JPEG (Joint Photographic Experts Group) [3] has been quite popular and influential. JPEG mainly employs discrete cosine transform (DCT) and entropy coding techniques to compress the images. While the JPEG compression method has gained widespread popularity, it does introduce visible artifacts including blurring, ringing and blocking [4]. JPEG2000 [5] is proposed forward to address the problems in JPEG. JPEG2000 adopts 2D wavelet transform and arithmetic coding to achieve higher compression efficiency.

Besides utilizing transforms and entropy coding techniques, a theory framework known as compressed sensing (CS) [6–8] was proposed to overcome the limitation that a signal must be sampled at the Nyquist sampling rate [9]. The CS theory has shed light on the problem of compression and reconstruction. Optimization techniques such as total variation (TV) minimization [10] and approximate message passing (AMP) [11] can be used in the recovery phase in the CS framework. TV minimization for image denoising was first introduced in [12]. TV minimization takes the advantage that it can better accurately preserve the edges or boundaries at certain compression ratios. In [13], the method “total variation minimization by augmented Lagrangian and alternating direction algorithms” (TVAL3) is proposed and has been used widely in image recovery problems. Comparisons in [14] suggest that the TVAL3 solver turns out to be fast and efficient so long as the reconstruction parameters are sufficient for a satisfying reconstruction. Meanwhile, based on the AMP [11] recovery algorithm, the D-AMP [15] algorithm is proposed to enhance CS recovery. In the scheme of D-AMP, the existing rich knowledge of signal denoiser is utilized to design the solver. Tests in [15] show that the D-AMP maintains a low computational footprint. Compressed sensing-based techniques have been explored in real-life scenarios like mine monitoring image compression [16] and landslide monitoring system [17]. The non-learning compressed sensing methods do achieve some success, but they struggle to produce sound recoveries at low compression ratios.

1.3. Data-driven Approaches

Due to the advancement of information technology, more data is within the reach of researchers. The data-driven approaches have found their way into various fields including signal processing [18], control systems [19–22] and especially vision tasks [23–27]. In particular, the deep learning-based method has stood out among the data-driven approaches. This section explores the recent development of deep learning-based image compression methods.

1.3.1. Convolution Neural Network based Image Compression

In more recent years, convolution neural networks (CNNs) has gained great attention due to the improvement of computing devices. As for image compression utilizing CNN, it generally involves designing image codecs with neural networks and constructing appropriate loss functions.

One genre of compression method combines the ideas of compressed sensing into CNN. For instance, the network DeepInverse proposed in [28] uses fully connected layers to simulate the compression process and stacks convolution layers for decompression. Back-propagation is applied to train the networks. This idea is extended further by ReconNet [29] which uses more convolution layers to attack the decompression problem. In [30], a deep residual reconstruction network is proposed

to recover images more accurately. However, this series of methods are more likely to blur edges in the recovered image especially at low compression ratios, according to the results reported by their authors.

Another genre of CNN based compression methods utilize the semantic information in images, since preserving semantic information will render the recovered image more eye-pleasing. Ballé et al. introduce an end-to-end optimized CNN image compression network in [31]. The method is based on non-linear coding rather than linear coding used by JPEG. One important contribution of [31] is that the authors propose an method which simulates the quantizer in the training procedure to deal with the problem of zero derivatives due to quantization. Li et al. point out that in [4] it is inappropriate to allocate the same number of codes for each spatial position in an image. They propose the importance map to guide the spatially variant bit allocation. To further compress the data, they introduce the convolutional entropy encoder to compress the binary codes and the importance map. In [32], the authors combine the deep-learning-based image semantic analysis into image compression as well. Unlike [4] which focuses more on the edge of objects, the method in [32] emphasizes the semantic analysis of the whole region. Results in their experiments show the method can improve the visual quality under the same compression overhead. However, it can be quite complicated to adjust the compression ratios of this genre of methods. Moreover, these methods are rarely applied at very low compression ratios.

1.3.2. Recurrent Neural Network Based Image Compression

Unlike the feed-forward CNN, the recurrent neural network (RNN) is state-aware. The output of an RNN is not only related to current input, but also the previous input. Lyu et al. propose to combine the knowledge of block-sparsity recovery into RNN deep learning in [33]. Their method acquires the spatial correlations between nonzero elements of block-sparse signals. It is applied to not only images but also audio data. However, the method proposed in [33] requires the input data to be sparse, which limits its compression capability. In [34], Toderici et al. combine the scaled-additive coding framework into RNN-based image compression scheme. The highlight in [34] is that the architectures proposed can provide variable compression rates during deployment without retraining the network. In [35], Minnen et al. propose a spatially adaptive image compression framework with quality-sensitive bit rate adaptation. However, though their method outperforms JPEG, it is still inferior to JPEG2000 [36].

1.3.3. Generative Adversarial Network Based Image Compression

Generative adversarial network (GAN) is another promising deep learning method developed during recent years. In the GAN scheme, a generator network and a discriminator network are optimized simultaneously. The discriminator network is trained to determine whether a sample is generated by the generator network, while the generator network needs to fool the discriminator into wrong decisions. In regards of image compression utilizing GAN, Ripple and Bourdev in [37] propose an architecture of autoencoder featuring pyramidal analysis, an adaptive coding module, and regularization of the expected code length. It produces images 2.5 times smaller than JPEG and JPEG2000, while achieving realtime performance using GPU. Jia et al. in [38] propose a light filed image compression framework driven by a GAN-based sub-aperture image generation and a cascaded hierarchical coding structure. Their method outperforms the state-of-the-art learning-based light field image compression approach with on average 4.9% BD-rate [39] reductions. In [40], Agustsson et al. propose a GAN-based framework targeting extremely low bitrate compression. Their method pushes the bitrate below 0.1 bpp while still achieves eye-pleasing results.

1.4. The Objectives and the Organization of the Paper

Considering the demand of image compression at very low compression ratios in underground mines, in this paper, we propose an image codec network based on CNN and a new loss function

based on discrete wavelet transform. The new loss function is dedicated to preserving edges in the images of underground mines. The remaining of the paper is organized as follows: Section 2 elaborates the proposed method by discussing the network architecture and the construction of the loss function. Section 3 provides experiments which demonstrate the performance and analysis of the proposed method. Section 4 concludes the paper with further discussion about the proposed method.

2. The Proposed Image Compression Method

2.1. Overview

Before introducing the network architecture, it is necessary to understand the workflow of the proposed compression method. As shown in Figure 1, a gray-scale image or one of the channels of an RGB color image is taken as the input. We view the input image as a matrix x . For simplicity, we assume the input image is square, which means x has the same number of rows and columns. The image matrix x is “vectorized” into one vector x_v by concatenating each row of the matrix. The encoder module compresses x_v to a feature vector y . Then the decoder module is applied to approximate x using the feature vector y . The approximation of x is denoted as \hat{x} . During training, both the recovered image and the original image are fed into the loss function. Back-propagation will try to minimize the value of loss by updating the weights in the encoder and decoder module.

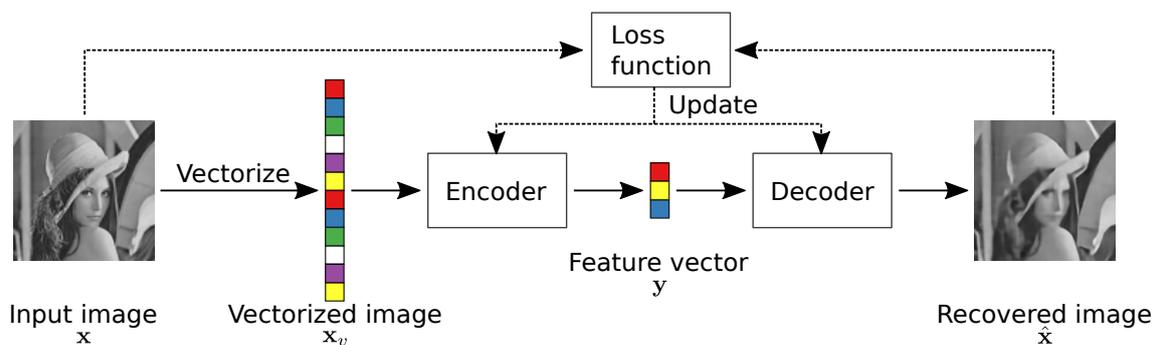


Figure 1. The workflow of the proposed method.

If there are N numbers in the image matrix x and M numbers in the feature vector y , then we define the compression ratio r as

$$r = M/N. \quad (1)$$

In short, the encoder module is responsible for compressing the image and determining the compression ratio, while the decoder module takes care of the recovery process.

2.2. The Network Architecture

2.2.1. The Encoder Module

The weight matrix \mathbf{W} of size $M \times N$ is multiplied by the “vectorized” image x_v . Then the product is added by the bias vector \mathbf{b} to derive the feature vector y :

$$y = \mathbf{W}x_v + \mathbf{b}. \quad (2)$$

In Equation (2), both the weight matrix \mathbf{W} and the bias vector \mathbf{b} are parameters to be learnt during back-propagation. \mathbf{W} is initialized using He initialization [41], while \mathbf{b} is initialized with zeros.

2.2.2. The Decoder Module

The network architecture of the decoder module is illustrated in Figure 2. The feature vector y is first upsampled to y' using nearest-neighbor interpolation [42]. The length of y' is determined by Equation (3):

$$\text{length}(y') = \lceil \sqrt{M} \rceil^2, \tag{3}$$

where M is the length of vector y . The symbol $\lceil z \rceil$ means rounding number z to the nearest integer more than or equal to z . The vector y' is then reshaped into the initial feature map F using Equation (4):

$$F[i, j] = y'[(i - 1) \times \lceil \sqrt{M} \rceil + j], 1 \leq i, j \leq \lceil \sqrt{M} \rceil. \tag{4}$$

Afterwards, the initial feature map F is convolved with 96 filters of size 3×3 . We empirically add a batch-normalization [43] layer after the first convolution layer to accelerate training. Then the feature maps go through several residual units. Some residual units are followed by nearest-neighbor upsampling operation as in Figure 2. Finally, the feature maps are convolved with one filter of size 1×1 to derive the recovered image \hat{x} .

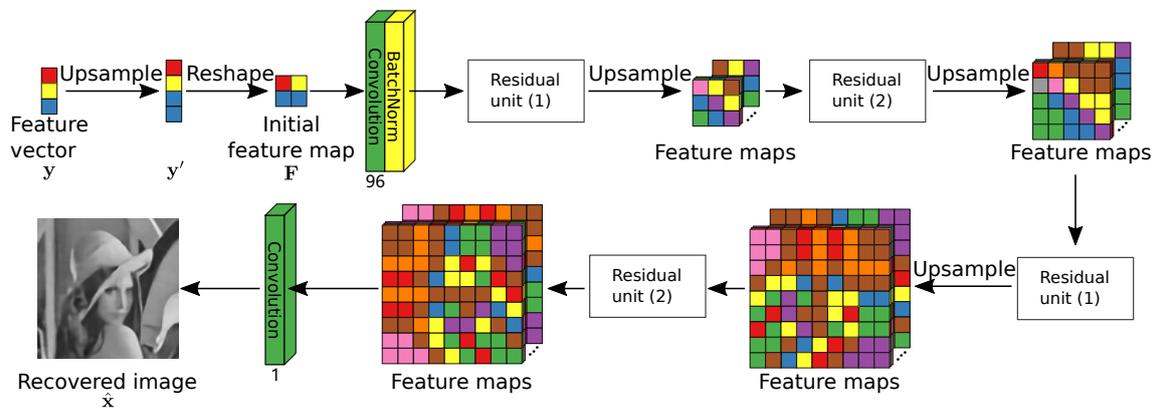


Figure 2. The network architecture of the decoder module.

The residual units. The introduction of residual units is inspired by [44]. As depicted in Figure 3, two types of residual units are used. Both types follow the two-branch connection pattern. The feature maps go through the two branches and add up at the output summator. The upper branches of the two types are identical. The lower branches differ in that residual unit (1) connects the input and the output with a stack of layers, but residual unit (2) connects the input and the output directly. Each convolution layer that appears in Figure 3 is composed of 96 filters of size 3×3 . After each convolution layer, there is a batch-normalization layer [43]. Each batch-normalization layer is then followed by a Leaky ReLU activation layer [45] if the batch-normalization layer is not directly connected to the output summator.

The nearest-neighbor upsampling operations. If the input image x is of size $n \times n$, then the second, third, and fourth upsampling operation in Figure 2 resize the feature maps to size $\frac{1}{2}n \times \frac{1}{2}n \times 96$, $\frac{3}{4}n \times \frac{3}{4}n \times 96$, and $n \times n \times 96$, respectively.

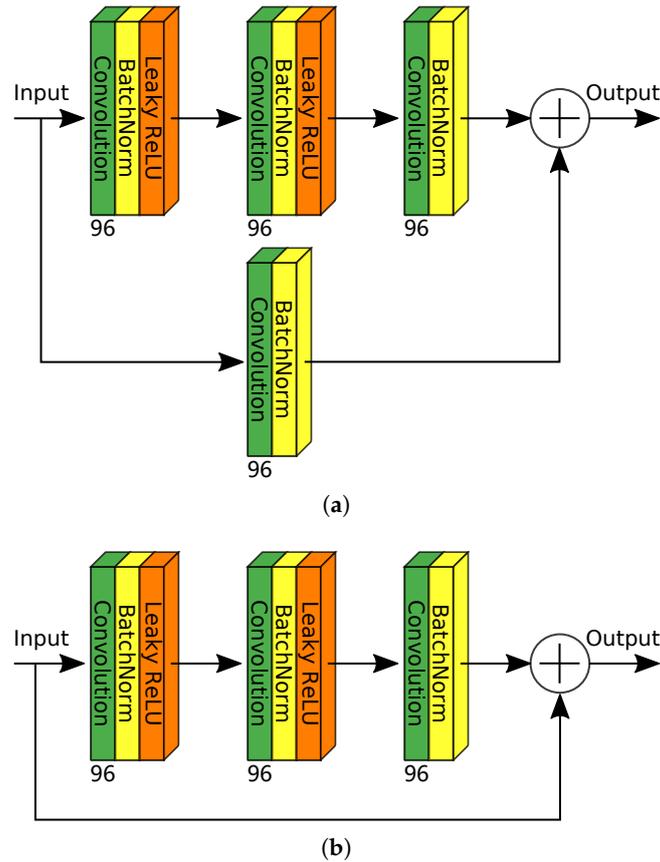


Figure 3. The residual units: (a) residual unit (1); (b) residual unit (2). All convolution layers in the two types of residual units employ filters of size 3×3 .

2.3. The Proposed Loss Function

2.3.1. Combination of Two Types of Loss Functions

Image recovering problems are conventionally seen as optimization problems that minimize the l_2 loss between the recovered and original image. However, from the perspective of image recovery quality assessment, l_2 metric does not reflect every aspect of signal fidelity [46]. Therefore, it is necessary to combine other metrics that compensate for what is missing in l_2 loss when constructing the loss function.

In this section, we propose a metric termed discrete wavelet structural similarity (DW-SSIM) that focuses the recovery of edges of the images. Our loss function is the weighted sum of DW-SSIM loss and l_2 loss:

$$L(\mathbf{x}, \hat{\mathbf{x}}) = \sum_{\mathbf{x} \in \Omega} (\beta_1 L_F(\mathbf{x}, \hat{\mathbf{x}}) + \beta_2 L_S(\mathbf{x}, \hat{\mathbf{x}}))$$

$$\beta_1 + \beta_2 = 1$$

$$0 \leq \beta_1, \beta_2 \leq 1,$$
(5)

where Ω represents a set of training image, $L_F(\mathbf{x}, \hat{\mathbf{x}})$ denotes the l_2 loss, $L_S(\mathbf{x}, \hat{\mathbf{x}})$ denotes the DW-SSIM loss, and $\beta_1 = 0.5$ and $\beta_2 = 0.5$ are weights. Both $L_F(\mathbf{x}, \hat{\mathbf{x}})$ and $L_S(\mathbf{x}, \hat{\mathbf{x}})$ are set up to fall in range $[0, 1)$. Section 2.3.2 will provide the expression of $L_F(\mathbf{x}, \hat{\mathbf{x}})$, while Section 2.3.3 will explain $L_S(\mathbf{x}, \hat{\mathbf{x}})$ in details.

2.3.2. l_2 Loss

We propose to use Frobenius norm in $L_F(\mathbf{x}, \hat{\mathbf{x}})$ to derive the l_2 loss:

$$L_F(\mathbf{x}, \hat{\mathbf{x}}) = \frac{\|\hat{\mathbf{x}} - \mathbf{x}\|_2^2}{\|\mathbf{x}\|_2^2}. \quad (6)$$

It is worth noting that the denominator of Equation (6) cannot be zero. However, since \mathbf{x} is taken from the natural images instead of artificial generated matrices, it is impossible for \mathbf{x} to be a zero matrix.

2.3.3. Discrete Wavelet Similarity (DW-SSIM) and DW-SSIM Loss

Inspired by structural similarity (SSIM) [47] and complex-wavelet structure similarity (CW-SSIM) [48], we propose to use two-dimensional discrete wavelet transform (2D-DWT) [49] to analyze the similarity between the recovered image and the original image. The similarity is termed DW-SSIM which stands for discrete-wavelet similarity.

2D-DWT. The 2D-DWT is able to decompose an image into different levels of subbands. The first level is the decomposition of the original image. Each level is composed of four subband images which can be referred to as low–low (LL), low–high (LH), high–low (HL) and high–high (HH). The LL image at each level can be further decomposed into the next level of subbands. The LH image represents the variation along the vertical direction, HL image the horizontal direction, and HH image the diagonal direction [49]. The high-frequency LH, HL, and HH subband images altogether form the details of the original image. As the decomposition level goes higher, the subband images become coarser, thus details of different scales can be analyzed. Figure 4 provides an example of a three-level 2D-DWT decomposition of an image.

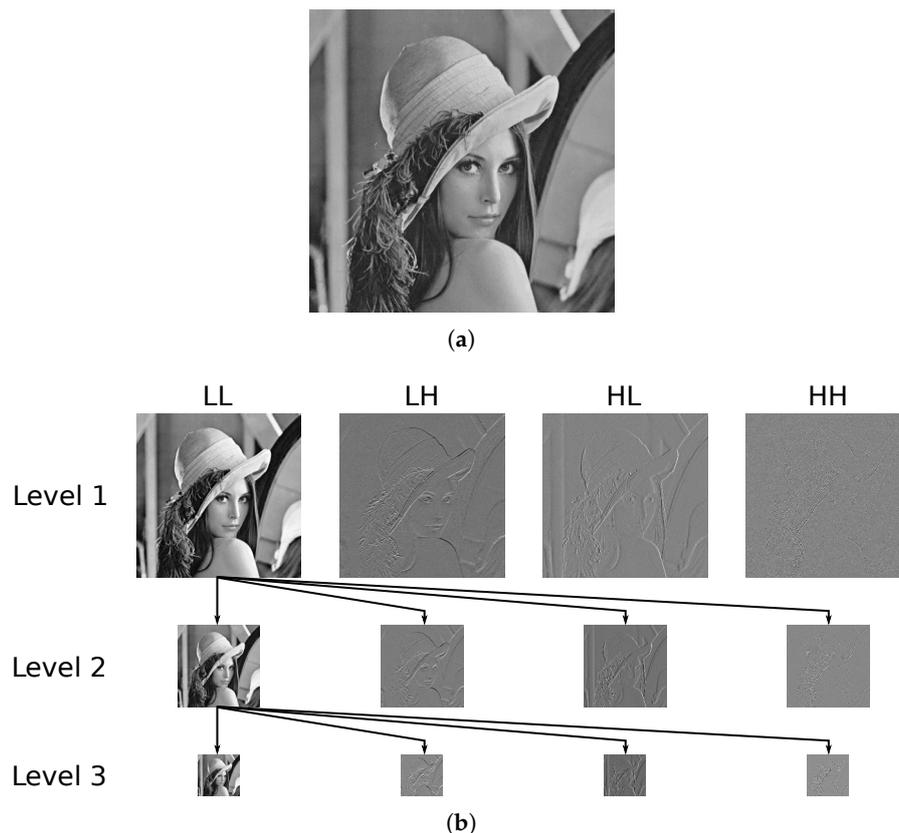


Figure 4. Illustration of 2D-discrete wavelet transform (DWT) image decomposition. (a) Original image. (b) Three-level decomposition of the image. For clarity, every intermediate low–low (LL) image is put in its place, yet DWT only preserves the LL image of the highest level.

DW-SSIM. We divide the calculation of DW-SSIM between the original image and the recovered image into two stages. The first stage involves figuring out the local DW-SSIM, where a “window” slides through the original image and the recovered image. 2D-DWT is performed on the image patches

within the “window” to derive the decomposition. We define the local low frequency DW-SSIM $S_{L,t}$ and high frequency DW-SSIM $S_{H,t}$ of the image patches as

$$S_{L,t}(\mathbf{c}^{(1)}, \mathbf{c}^{(2)}) = \frac{2 \left| \sum_u \sum_v \mathbf{c}_{LL}^{(1)}[u, v] \mathbf{c}_{LL}^{(2)}[u, v] \right| + K}{\sum_u \sum_v \left| \mathbf{c}_{LL}^{(1)}[u, v] \right|^2 + \sum_u \sum_v \left| \mathbf{c}_{LL}^{(2)}[u, v] \right|^2 + K}, \quad (7)$$

$$S_{H,t}(\mathbf{c}^{(1)}, \mathbf{c}^{(2)}) = \frac{1}{J} \sum_{j=1}^J \frac{2 \left| \sum_i \sum_u \sum_v \mathbf{c}_i^{(1)}[u, v] \mathbf{c}_i^{(2)}[u, v] \right| + K}{\sum_i \sum_u \sum_v \left| \mathbf{c}_i^{(1)}[u, v] \right|^2 + \sum_i \sum_u \sum_v \left| \mathbf{c}_i^{(2)}[u, v] \right|^2 + K}, i \in \{LH_j, HL_j, HH_j\}. \quad (8)$$

In Equations (7) and (8), K is a small positive constant for arithmetic robustness and K is set to 0.01. $\mathbf{c}^{(1)}$ and $\mathbf{c}^{(2)}$ refer to the corresponding subband images of the original image patch and the recovered image patch after 2D-DWT, respectively. The wavelet function we use is the Haar wavelet. t is the patch index. $J = 3$ is the maximum decomposition level, and $c_{LH_j}, c_{HL_j}, c_{HH_j}$ are high frequency subband images at the j -th level.

To better understand Equation (7), one can ignore K , “vectorize” (as in Section 2.2.1) \mathbf{c} into \mathbf{c}_v and rewrite it as

$$\begin{aligned} S_{L,t}(\mathbf{c}^{(1)}, \mathbf{c}^{(2)}) &= S_{L,t}(\mathbf{c}_v^{(1)}, \mathbf{c}_v^{(2)}) = 2 \frac{|\mathbf{c}_v^{(1)} \cdot \mathbf{c}_v^{(2)}|}{\|\mathbf{c}_v^{(1)}\|_2^2 + \|\mathbf{c}_v^{(2)}\|_2^2} \\ &= 2 \frac{\|\mathbf{c}_v^{(1)}\|_2 \|\mathbf{c}_v^{(2)}\|_2}{\|\mathbf{c}_v^{(1)}\|_2^2 + \|\mathbf{c}_v^{(2)}\|_2^2} \left| \frac{\mathbf{c}_v^{(1)} \cdot \mathbf{c}_v^{(2)}}{\|\mathbf{c}_v^{(1)}\|_2 \|\mathbf{c}_v^{(2)}\|_2} \right| \\ &= 2 \left(\frac{\|\mathbf{c}_v^{(1)}\|_2}{\|\mathbf{c}_v^{(2)}\|_2} + \frac{\|\mathbf{c}_v^{(2)}\|_2}{\|\mathbf{c}_v^{(1)}\|_2} \right)^{-1} |\cos(\theta)|. \end{aligned} \quad (9)$$

In Equation (9), the first term is determined by the energy of the subband images. It will reach its maximum value 1 only if $\|\mathbf{c}_v^{(1)}\|_2 = \|\mathbf{c}_v^{(2)}\|_2$. In the second term, $\cos(\theta) = \frac{\mathbf{c}_v^{(1)} \cdot \mathbf{c}_v^{(2)}}{\|\mathbf{c}_v^{(1)}\|_2 \|\mathbf{c}_v^{(2)}\|_2}$ is the cosine similarity [50]. If $\mathbf{c}_v^{(1)}$ and $\mathbf{c}_v^{(2)}$ point to roughly the same direction, the cosine similarity will be close to 1. However, the cosine function falls in range $[-1, 1]$. Therefore we are taking the absolute value so that it falls in $[0, 1]$. The interpretation of Equation (8) is largely the same with that of Equation (7). Equation (8) additionally averages the contribution of each level of subband to the high frequency DW-SSIM in order to cope with the patterned noise in underground mine images. This can be better understood through the discussion in Section 3.3.

In the second stage, a weighted sum of $S_{H,t}$ and $S_{L,t}$ is figured out to form the final DW-SSIM S :

$$S(\mathbf{x}, \hat{\mathbf{x}}) = \frac{1}{T} \sum_{t=1}^T (\gamma_1 S_{L,t} + \gamma_2 S_{H,t}), \quad (10)$$

where T is the total number of image patches, γ_1 and γ_2 are parameters to adjust the weight of low frequency subband and high frequency subbands. Since we want to emphasize high frequency details such as edges and spikes in the image, we set $\gamma_1 = 0.2$ and $\gamma_2 = 0.8$.

The computation of DW-SSIM is summarized with Algorithm 1. The window length l in the proposed method is set to 15. The stride s that the window will move in each iteration is set to 8.

Algorithm 1: The procedure to compute discrete wavelet similarity (DW-SSIM).

Input: The original image img-ori and the recovered image img-rec of the same height H and width W ($H > 0, W > 0$); the decomposition level J ; the stride s that the window will move in each iteration; the window length l ; the weights γ_1 and γ_2 in Equation (10)

Output: The DW-SSIM similarity S between img-ori and img-rec

```

 $S \leftarrow 0$ ;
 $t \leftarrow 0$ ;
 $\text{up} \leftarrow 0$ ;
 $\text{left} \leftarrow 0$ ;
while  $\text{up} < H$  do
   $\text{down} \leftarrow \text{up} + l$ ;
  while  $\text{left} < W$  do
     $t \leftarrow t + 1$ ;
     $\text{right} \leftarrow \text{left} + l$ ;
    Get image patch  $\text{patch-ori}$  within the window  $[\text{up}, \text{down}, \text{left}, \text{right}]$  from  $\text{img-ori}$ ;
    Get image patch  $\text{patch-rec}$  within the window  $[\text{up}, \text{down}, \text{left}, \text{right}]$  from  $\text{img-rec}$ ;
    Derive  $\mathbf{c}^{(1)}$  by performing 2D-DWT on  $\text{patch-ori}$ ;
    Derive  $\mathbf{c}^{(2)}$  by performing 2D-DWT on  $\text{patch-rec}$ ;
    Derive  $S_{L,t}$  from Equation (7);
    Derive  $S_{H,t}$  from Equation (8);
     $S \leftarrow S + \gamma_1 S_{L,t} + \gamma_2 S_{H,t}$ ;
     $\text{left} \leftarrow \text{left} + s$ ;
  end
   $\text{up} \leftarrow \text{up} + s$ ;
end
 $S \leftarrow S/t$ ;
return  $S$ ;

```

DW-SSIM loss. The DW-SSIM defined in Equation (10) falls in range $(0, 1]$. The more the original image and the recovered image matches each other, the closer DW-SSIM S is to 1. However, the loss should be near 0 if the model has done a perfect recovery. Moreover, the loss should fall in range $[0, 1)$. Therefore, we define the DW-SSIM loss as:

$$L_S(\mathbf{x}, \hat{\mathbf{x}}) = 1 - S(\mathbf{x}, \hat{\mathbf{x}}). \quad (11)$$

2.4. Learning the Parameters

The encoder module and the decoder module can be trained in an end-to-end manner using the proposed network architecture and the proposed loss function. Mini-batch gradient descent is used to train the model with the batch size being 64. The Adam [51] optimizer is utilized as well. We set the initial learning rate to 5×10^{-4} . The learning rate is multiplied by 0.2 when the loss is not going down during training. The training is stopped if the learning rate drops below 1×10^{-6} .

3. Results

3.1. Overview

In order to generalize the recovery capability, the network of the proposed method is trained on both images from video images we have collected in underground mines and images from the COCO 2014 dataset [52]. We build the training set by extracting the 100×100 center-crop patches from the images, and converting them to grayscale images.

After the model is trained, test images (as in Figure 5) are passed to the model to perform the compression and recovery. We test our method on both standard images of Barbara, Fingerprint, and Lena to verify its effectiveness. In addition, we test the proposed method on images of coal cutter and tunnel boring machine (TBM) which are from real underground mines to evaluate the performance in the application-specific environment.

The recovery quality is quantitatively evaluated with peak-signal-to-noise ratio (PSNR) and structural similarity (SSIM) [46]:

$$\text{PSNR}(\mathbf{x}, \hat{\mathbf{x}}) = 10 \log_{10} \frac{d^2}{\frac{1}{N} \sum_{i=1}^N (\mathbf{x}[i] - \hat{\mathbf{x}}[i])^2}, \quad (12)$$

$$\text{SSIM}(\mathbf{x}, \hat{\mathbf{x}}) = \left(\frac{2\mu_{\mathbf{x}}\mu_{\hat{\mathbf{x}}} + C_1}{\mu_{\mathbf{x}}^2 + \mu_{\hat{\mathbf{x}}}^2 + C_1} \right) \cdot \left(\frac{2\sigma_{\mathbf{x}}\sigma_{\hat{\mathbf{x}}} + C_2}{\sigma_{\mathbf{x}}^2 + \sigma_{\hat{\mathbf{x}}}^2 + C_2} \right) \cdot \left(\frac{\sigma_{\mathbf{x}\hat{\mathbf{x}}} + C_3}{\sigma_{\mathbf{x}}\sigma_{\hat{\mathbf{x}}} + C_3} \right). \quad (13)$$

In Equation (12), d is the dynamic range of pixel intensities, and N is the number of pixels in the image. In Equation (13), $\mu_{\mathbf{x}}$ and $\mu_{\hat{\mathbf{x}}}$ are means of \mathbf{x} and $\hat{\mathbf{x}}$, and $\sigma_{\mathbf{x}}^2$ and $\sigma_{\hat{\mathbf{x}}}^2$ are variances of \mathbf{x} and $\hat{\mathbf{x}}$. $\sigma_{\mathbf{x}\hat{\mathbf{x}}}$ is the cross correlation of \mathbf{x} and $\hat{\mathbf{x}}$. The small positive constants $C_1 = C_2 = C_3 = 0.01$ prevent numerical instability of each term.

To verify the effectiveness of the proposed method, the quantitative evaluation at compression ratios of 0.25, 0.20, 0.15, 0.10, 0.04 and 0.01 is carried out, with the compression ratio defined in Equation (1). In addition, the proposed method is compared to the algorithms of D-AMP [15], ReconNet [13] and TVAL3 [29] at different compression ratios. For simplicity, we do not re-implement the algorithms but use the demo code provided by the authors' websites instead.

Further, visual quality evaluation of recovery is presented at some specific compression ratios.

Finally, the robustness of the proposed method is tested by recovering images contaminated by different levels of Gaussian noise.

The proposed method was implemented with Pytorch [53] and pytorch_wavelet package (https://github.com/fbcotter/pytorch_wavelets). The training process is carried out on Ubuntu 18.04.2, with Nvidia Tesla K80 GPU and Intel Xeon CPU. More details about the implementation can be found in the code which we have made public on the Internet (<https://github.com/y0umu/ResCSNet>).



Figure 5. The test images: (a) Barbara; (b) Fingerprint; (c) Lena; (d) Coal cutter; (e) Tunnel boring machine (TBM).

3.2. Quantitative Evaluation

Tables 1 and 2 provide quantitative measurements of the proposed method and other algorithms at different compression ratios. As the compression ratio r decreases, all the algorithms being compared have PSNR and SSIM decreased. It can be interpreted from Table 1 that the proposed method is second only to D-AMP at compression ratio $r \geq 0.20$ for both standard test images and real underground mine images. Yet the proposed method achieves the highest PSNR compared to other algorithms at a compression ratio $r \leq 0.15$. It should be also noted that for the recoveries of images of coal cutter and TBM at compression ratios $r \leq 0.04$, the proposed method has an edge over other algorithms by a margin of at least 1.8 dB, indicating the potential of the application-specific usage in mines of the proposed method.

Table 1. Peak signal-to-noise ratio (PSNR) (in dB) comparison for different algorithms on test images. r is the compression ratio.

Image	Algorithm	$r = 0.25$	$r = 0.20$	$r = 0.15$	$r = 0.10$	$r = 0.04$	$r = 0.01$
Barbara	D-AMP	26.61	25.37	24.00	21.73	15.37	7.23
	ReconNet	25.14	22.80	21.41	21.79	19.74	16.20
	TVAL3	22.40	21.28	19.76	18.87	16.19	15.15
	DR ² -Net	25.43	21.64	19.86	20.99	18.34	16.08
	Proposed	27.23	27.62	26.50	24.15	21.76	17.86
Fingerprint	D-AMP	20.99	20.64	19.41	19.07	11.65	5.24
	ReconNet	17.56	17.20	17.25	16.68	16.10	15.55
	TVAL3	18.25	17.45	17.04	15.57	14.08	9.68
	DR ² -Net	18.30	16.57	15.98	17.16	16.26	15.20
	Proposed	19.76	19.80	19.70	19.39	19.17	18.68
Lena	D-AMP	30.28	28.40	26.57	24.38	11.71	6.57
	ReconNet	23.83	22.65	21.58	20.32	18.50	15.90
	TVAL3	21.26	20.68	19.51	17.81	16.37	15.17
	DR ² -Net	26.37	21.93	20.02	21.82	19.07	15.77
	Proposed	28.82	29.01	28.44	25.48	24.08	19.47
Coal cutter	D-AMP	21.81	21.86	20.90	19.10	14.36	8.14
	ReconNet	18.78	18.35	17.67	17.24	16.26	14.52
	TVAL3	12.52	10.94	9.87	8.17	10.48	12.50
	DR ² -Net	20.22	17.71	16.65	17.76	16.19	14.78
	Proposed	21.78	21.84	21.40	20.05	18.08	17.34
TBM	D-AMP	29.68	28.02	26.30	24.51	17.63	8.76
	ReconNet	23.89	22.95	22.13	21.21	19.24	17.65
	TVAL3	17.27	16.17	14.88	14.35	13.16	13.71
	DR ² -Net	25.65	22.11	20.87	22.04	19.50	17.53
	Proposed	27.67	27.27	27.12	24.95	22.46	20.03

Table 2. SSIM comparison for different algorithms on test images. r is the compression ratio.

Image	Algorithm	$r = 0.25$	$r = 0.20$	$r = 0.15$	$r = 0.10$	$r = 0.04$	$r = 0.01$
Barbara	D-AMP	0.8570	0.7781	0.7583	0.6189	0.0624	0.0129
	ReconNet	0.7449	0.7037	0.6062	0.5506	0.3805	0.2226
	TVAL3	0.7391	0.6834	0.6154	0.4692	0.3134	0.2281
	DR ² -Net	0.8165	0.7396	0.6774	0.6137	0.3947	0.2283
	Proposed	0.8823	0.8950	0.8648	0.7832	0.6087	0.2859
Fingerprint	D-AMP	0.5530	0.4063	0.2709	0.2288	0.1050	0.0029
	ReconNet	0.2438	0.2245	0.1890	0.1871	0.1412	0.0970
	TVAL3	0.3448	0.2884	0.2496	0.1948	0.1339	0.0774
	DR ² -Net	0.3030	0.2291	0.2115	0.2044	0.1484	0.0976
	Proposed	0.3464	0.3742	0.3307	0.2871	0.2103	0.1498
Lena	D-AMP	0.8867	0.8667	0.8174	0.7550	0.4343	0.0235
	ReconNet	0.7412	0.7084	0.6436	0.5997	0.4558	0.3181
	TVAL3	0.7420	0.7145	0.6596	0.5370	0.3735	0.2869
	DR ² -Net	0.8200	0.7771	0.7052	0.6597	0.5119	0.3352
	Proposed	0.8930	0.9040	0.8879	0.8301	0.7437	0.4440
Coal cutter	D-AMP	0.6854	0.6793	0.6376	0.5148	0.1735	0.0363
	ReconNet	0.5470	0.4947	0.4377	0.4267	0.3371	0.2431
	TVAL3	0.3830	0.3146	0.2574	0.1838	0.2110	0.1608
	DR ² -Net	0.6358	0.5482	0.4672	0.4899	0.3467	0.2634
	Proposed	0.7320	0.7476	0.7049	0.6303	0.4923	0.3498
TBM	D-AMP	0.8711	0.6793	0.8027	0.7187	0.2829	0.0634
	ReconNet	0.7728	0.7319	0.6771	0.6522	0.5460	0.4318
	TVAL3	0.5445	0.4868	0.5069	0.4127	0.3794	0.3372
	DR ² -Net	0.8184	0.7523	0.7171	0.6755	0.5714	0.4456
	Proposed	0.8793	0.8764	0.8639	0.8058	0.6921	0.5359

From Table 2, it can be learned that the proposed method achieves the highest SSIM at every compression ratio for all the images except the Fingerprint image. Since the SSIM metric describes structural similarity between the recovered and the original images, it can be drawn to the conclusion that the proposed method preserves specific characteristics of the images better.

3.3. Visual Quality Evaluation

Figures 6 and 7 illustrate the recovered images of the proposed method and the algorithms being compared. The green boxes zoom in the image patches within the red boxes so that the details can be viewed clearly. As can be seen in most of the pictures, the proposed method recovers sharper edges with less blurring compared to other algorithms. In Figure 7 where the compression ratio is relatively low, the edges can still be discerned in the recovered image of the proposed method, while other recoveries tend to be more blurred. Combined with Tables 1 and 2, it can be found that the characteristic which the proposed method preserves is the edges in the image.

Figures 6 and 7 also demonstrate an interesting phenomenon. In the recovery of the Fingerprint image, the proposed method fails to recover the details either at a compression ratio $r = 0.15$ or $r = 0.04$. This is intended behavior and actually the proposed method deliberately “blurs” dense patterns in the recovered images to cope with the noise which is often seen in underground mine images. To explain the rationale behind this, suppose we take the image patches of size 15×15 at the same location from the recovered image and the original image of Fingerprint. Then 3-level 2D-DWT is applied on both patches and it can be discovered that the level 2 or level 3 subband images are almost identical. The major difference of the subbands lies in the level 1 decomposition. Recall that in Equation (8) each level is given the same significance, the difference between the recovered and original patch in level 1 decomposition is in effect “averaged out”. Therefore the DW-SSIM loss of the original dense patterned patch and the recovered blurred patch will be small, leading the proposed network to learn to blur the dense patterns.

3.4. Robustness against Noise

Since the tests in previous sections indicate that the proposed method takes an advantage when the compression ratio is low, we then test the noise robustness of the proposed method at a compression ratio $r = 0.04$ in this section. As depicted in Figures 8 and 9, Gaussian noise is added to the Lena and TBM test images to simulate the dusty environment in underground mines. The noise is zero-mean. The standard deviation σ of the noise is set to 5, 10, 15, 20, 25 and 30 to emulate different levels of noise. The noise-contaminated images are compressed at ratio $r = 0.04$. Then the similarity of the recovered images between the original test images is evaluated using the PSNR and SSIM measurement.

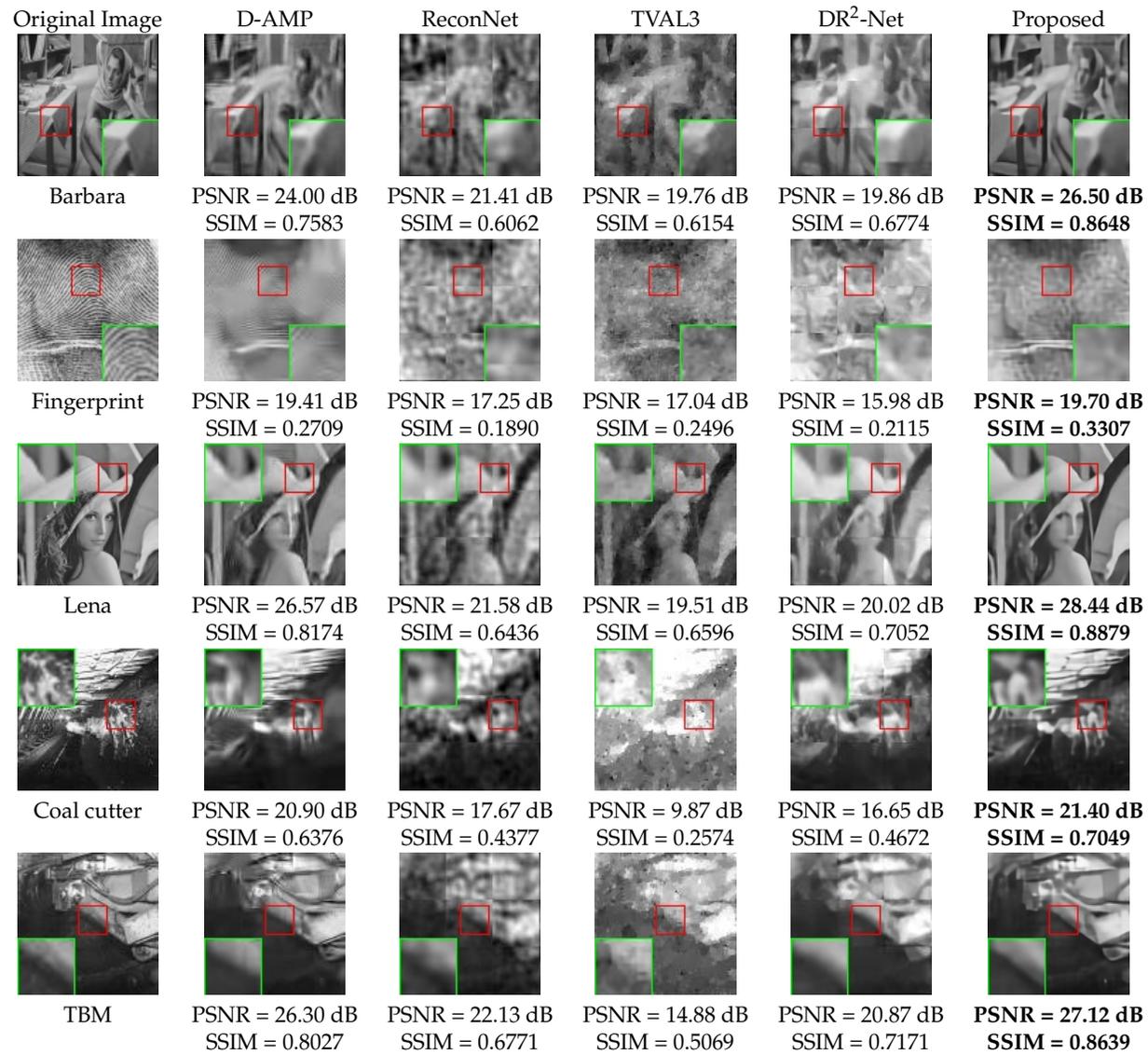


Figure 6. The recovered images at compression ratio $r = 0.15$.



Figure 7. The recovered images at compression ratio $r = 0.04$.

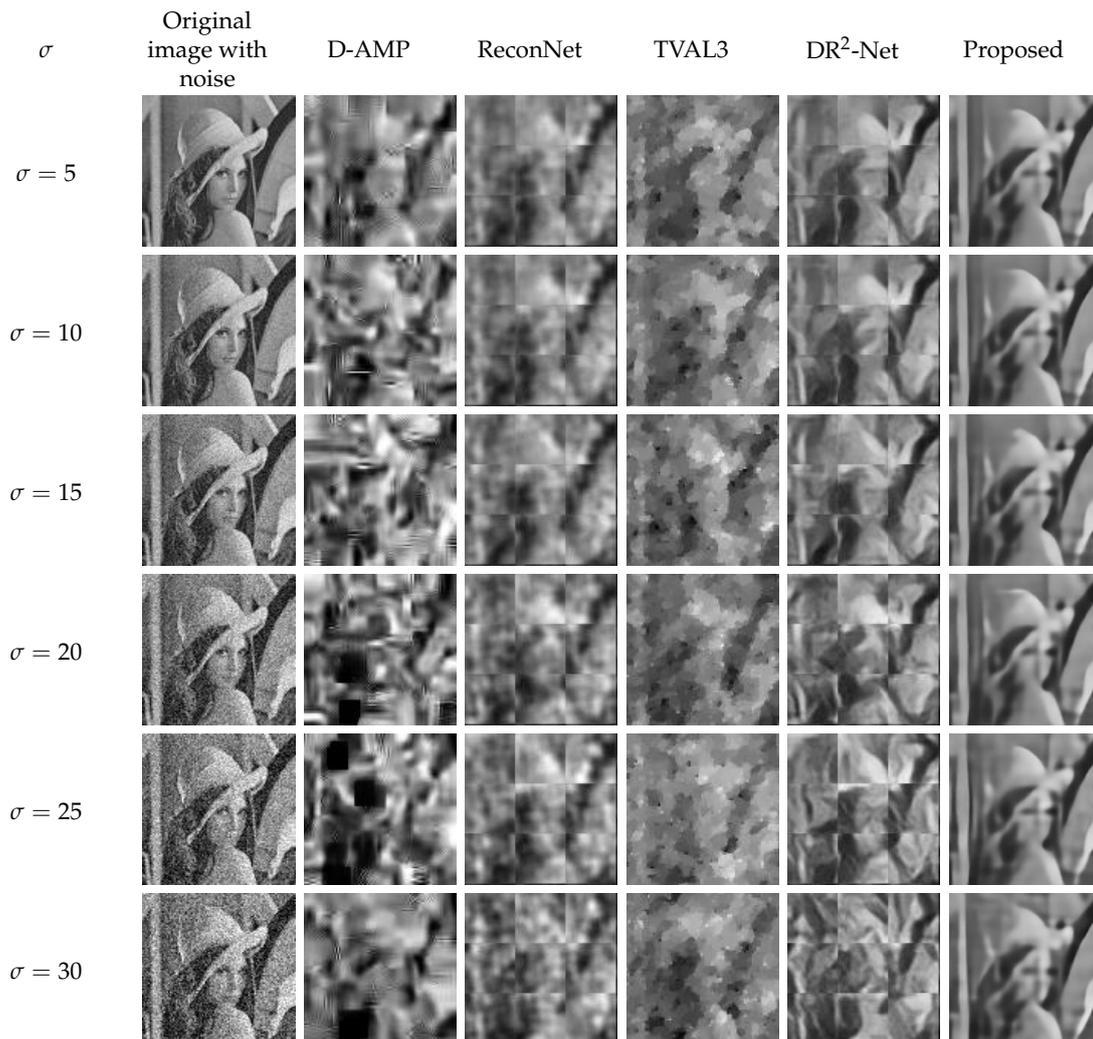


Figure 8. Comparison of recoveries of the Lena image at the presence of noise. σ denotes the standard deviation of the noise. The compression ratio r is 0.04.

As in Figure 8 and Figure 9, at all noise levels, fewer artifacts can be seen yet sharp edges are preserved in the recovered images of the proposed method. Further, Figure 10 plots the PSNR and SSIM curves as σ varies. The PSNR and SSIM of all algorithms drop as σ increases, yet PSNR and SSIM of the proposed method are higher than those of the algorithms being compared. As σ grows from 5 to 30, the decrease of PSNR and SSIM of the proposed method, which is no more than 1.6 dB and 0.11, is the least among the algorithms. Therefore, it can be concluded that the proposed method features noise robustness when the compression ratio is low.

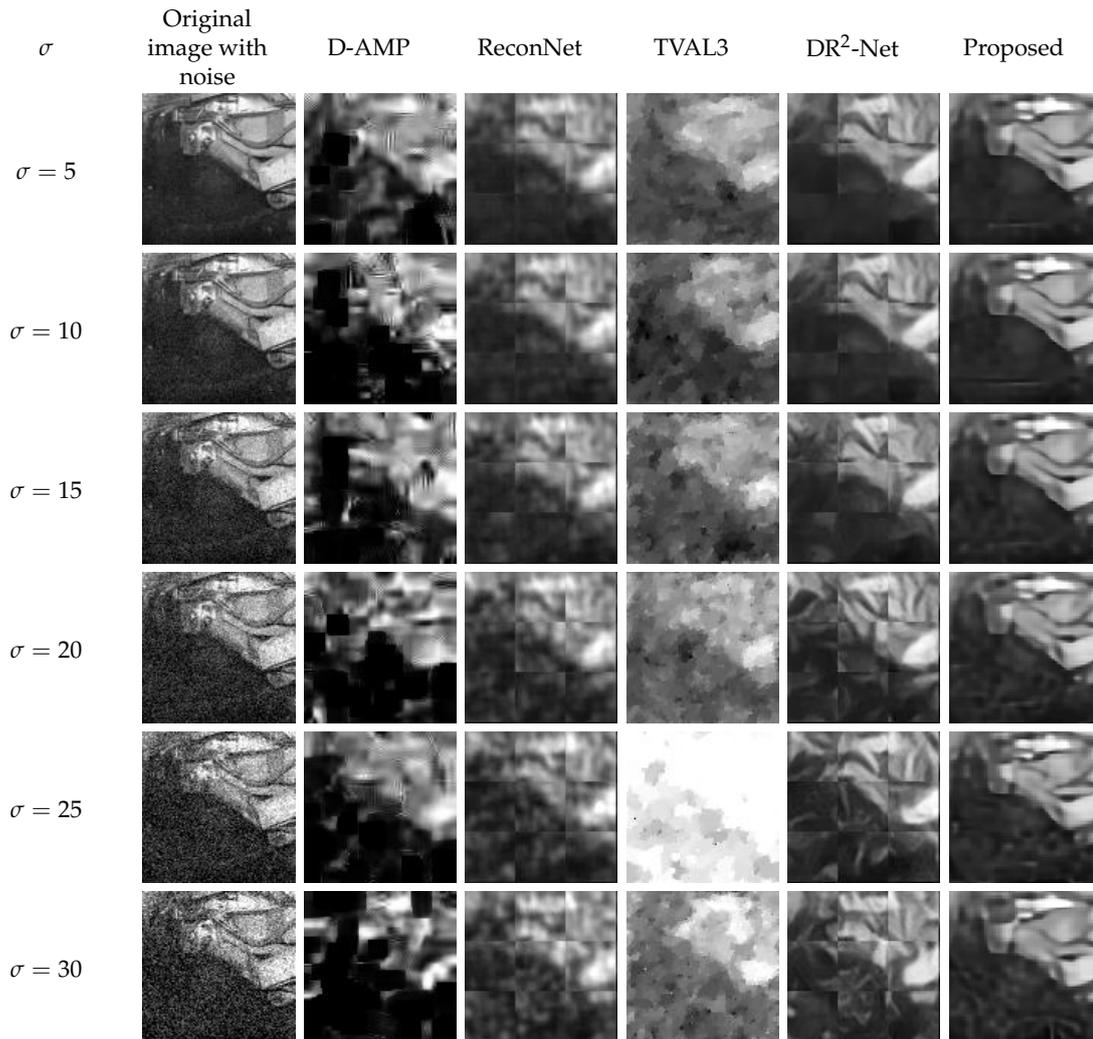
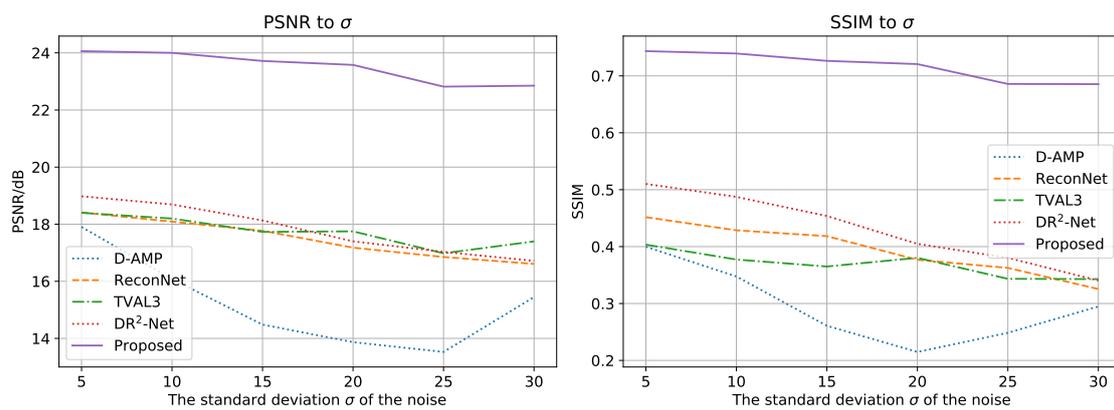


Figure 9. Comparison of recoveries of the TBM image at the presence of noise. σ denotes the standard deviation of the noise. The compression ratio r is 0.04.



(a)

Figure 10. Cont.

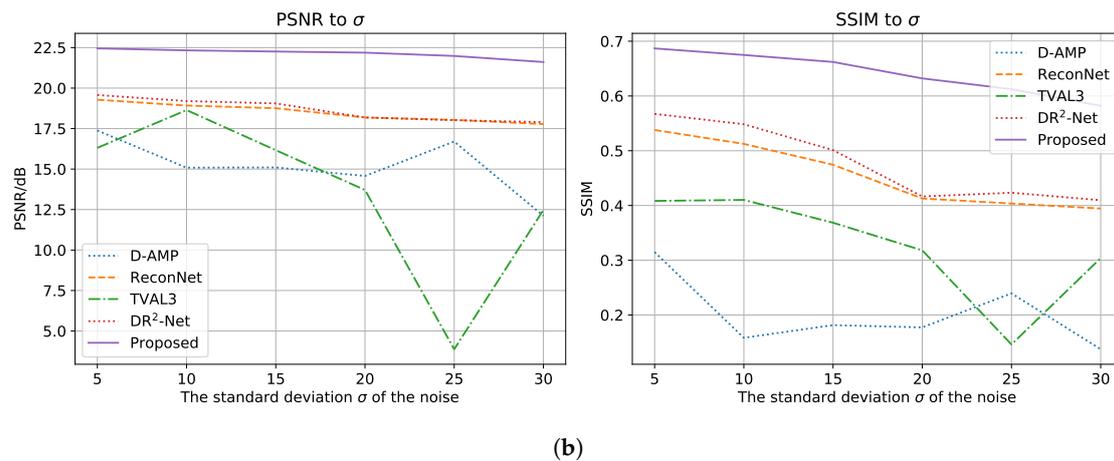


Figure 10. Plots of PSNR and SSIM against σ for the recoveries of noise contaminated (a) Lena image, (b) TBM image. The PSNR and SSIM are checked between the original test image (no noise added) and the recovered images. The compression ratio r is 0.04.

4. Conclusions

In this paper, we propose a CNN based image codec network which acts as the basis for the compression and recovery of images. We also propose a novel loss function that combines the knowledge of discrete wavelet transform to attack the problem of edge blurring in the recovered images. The proposed method is more suitable for the compression and recovery of underground mine images in that:

- The proposed method recovers sharp edges in the images. For underground mines, edges in the image are the key component to distinguish the foreground and background. By determining the boundaries of miners and equipment, it is possible for further image analysis to carry out.
- The proposed method features noise robustness. By blurring the dense patterns, the proposed method can filter out the noise especially seen in underground mines.
- Compared to other algorithms, the proposed method excels at low compression ratios. General image compression methods tend to strike a balance between the compression ratio and the recovery quality. They do not have to work at extremely low compression ratios as the transmission bandwidth available is comparably high. However, the proposed method is designed to work at low compression ratios to adapt to the harsh communication environment in underground mines.

In future work, we will combine other denoising techniques into the work presented in this paper as an attempt to achieve noise robustness without blurring the patterned areas. The current design of the DW-SSIM loss is not perfect in that the merits of cosine similarity is not fully preserved. Thus it is worth further investigating into the design of loss function. We will also train the model on other datasets in order to expand the application of the proposed method.

Author Contributions: Conceptualization, F.Z. and W.C.; data curation, Z.X.; formal analysis, F.Z. and Z.Z.; funding acquisition, F.Z. and W.C.; investigation, Z.X., H.Z. and J.L.; methodology, Z.X. and Z.Z.; project administration, F.Z. and W.C.; resources, F.Z.; software, Z.X., H.Z. and J.L.; Supervision, F.Z. and W.C.; validation, Z.X., W.C. and C.L.; visualization, Z.X.; writing—original draft, Z.X.; writing—review and editing, F.Z., Z.X., W.C. and C.L.

Funding: This research was funded by Foundation of the National Key Research and Development Program grant number 2016YFC0801800, National Natural Science Foundation of China grant number 51874300, National Natural Science Foundation of China and Shanxi Provincial People's Government Jointly Funded Project of China for Coal Base and Low Carbon grant number U1510115, and the Open Research Fund of Key Laboratory of Wireless Sensor Network and Communication, Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences grant numbers 20190902 and 20190913.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. Dai, S.; Finkelman, R.B. Coal as a promising source of critical elements: Progress and future prospects. *Int. J. Coal Geol.* **2018**, *186*, 155–164, doi:10.1016/j.coal.2017.06.005. [[CrossRef](#)]
2. Dohare, Y.S.; Maity, T.; Das, P.S.; Paul, P.S. Wireless Communication and Environment Monitoring in Underground Coal Mines- Review. *IETE Tech. Rev.* **2015**, *32*, 140–150, doi:10.1080/02564602.2014.995142. [[CrossRef](#)]
3. Wallace, G. The JPEG still picture compression standard. *IEEE Trans. Consum. Electron.* **1992**, *38*, xviii–xxxiv, doi:10.1109/30.125072. [[CrossRef](#)]
4. Li, M.; Zuo, W.; Gu, S.; Zhao, D.; Zhang, D. Learning Convolutional Networks for Content-Weighted Image Compression. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 3214–3223, doi:10.1109/CVPR.2018.00339. [[CrossRef](#)]
5. Marcellin, M.W.; Gormish, M.J.; Bilgin, A.; Boliek, M.P. An overview of JPEG-2000. In Proceedings of the Data Compression Conference (DCC 2000), Snowbird, UT, USA, 28–30 March 2000; pp. 523–541, doi:10.1109/DCC.2000.838192. [[CrossRef](#)]
6. Candes, E.; Romberg, J.; Tao, T. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Inf. Theory* **2006**, *52*, 489–509, doi:10.1109/TIT.2005.862083. [[CrossRef](#)]
7. Candès, E.J.; Romberg, J.K.; Tao, T. Stable signal recovery from incomplete and inaccurate measurements. *Commun. Pure Appl. Math.* **2006**, *59*, 1207–1223, doi:10.1002/cpa.20124. [[CrossRef](#)]
8. Donoho, D.L.; Tanner, J. Thresholds for the Recovery of Sparse Solutions via L1 Minimization. In Proceedings of the IEEE 2006 40th Annual Conference on Information Sciences and Systems, Princeton, NJ, USA, 22–24 March 2006; pp. 202–206, doi:10.1109/CISS.2006.286462. [[CrossRef](#)]
9. Joshi, A.M.; Sahu, C.; Ravikumar, M.; Ansari, S. Hardware implementation of compressive sensing for image compression. In Proceedings of the TENCON 2017—2017 IEEE Region 10 Conference, Penang, Malaysia, 5–8 November 2017; IEEE: Penang, Malaysia, 2017; Volume 2017-Decem, pp. 1309–1314, doi:10.1109/TENCON.2017.8228060. [[CrossRef](#)]
10. Chambolle, A. An Algorithm for Total Variation Minimization and Applications. *J. Math. Imaging Vis.* **2004**, *20*, 89–97, doi:10.1023/B:JMIV.0000011321.19549.88. [[CrossRef](#)]
11. Donoho, D.L.; Maleki, A.; Montanari, A. Message-passing algorithms for compressed sensing. *Proc. Natl. Acad. Sci. USA* **2009**, *106*, 18914–18919, doi:10.1073/pnas.0909892106. [[CrossRef](#)]
12. Rudin, L.I.; Osher, S.; Fatemi, E. Nonlinear total variation based noise removal algorithms. *Phys. D Nonlinear Phenom.* **1992**, *60*, 259–268, doi:10.1016/0167-2789(92)90242-F. [[CrossRef](#)]
13. Li, C.; Yin, W.; Jiang, H.; Zhang, Y. An efficient augmented Lagrangian method with applications to total variation minimization. *Comput. Optim. Appl.* **2013**, *56*, 507–530, doi:10.1007/s10589-013-9576-1. [[CrossRef](#)]
14. Kong, Q.; Gong, R.; Liu, J.; Shao, X. Investigation on Reconstruction for Frequency Domain Photoacoustic Imaging via TVAL3 Regularization Algorithm. *IEEE Photonics J.* **2018**, *10*, 1–15, doi:10.1109/JPHOT.2018.2869815. [[CrossRef](#)]
15. Metzler, C.A.; Maleki, A.; Baraniuk, R.G. From Denoising to Compressed Sensing. *IEEE Trans. Inf. Theory* **2016**, *62*, 5117–5144, doi:10.1109/TIT.2016.2556683. [[CrossRef](#)]
16. Zhao, X.; Shen, X.; Wang, K.; Li, W. A DCVS Reconstruction Algorithm for Mine Video Monitoring Image Based on Block Classification. *Preprints* **2018**, 2018070222, doi:10.20944/PREPRINTS201807.0222.V1. [[CrossRef](#)]
17. Wang, Y.; Liu, Z.; Wang, D.; Li, Y.; Yan, J. Anomaly detection and visual perception for landslide monitoring based on a heterogeneous sensor network. *IEEE Sens. J.* **2017**, *17*, 1, doi:10.1109/JSEN.2017.2704584. [[CrossRef](#)]
18. Qiao, X.; Yang, F.; Zheng, J. Ground Penetrating Radar Weak Signals Denoising via Semi-soft Threshold Empirical Wavelet Transform. *Ingénierie Des Systèmes d'Information* **2019**, *24*, 207–213, doi:10.18280/isi.240213. [[CrossRef](#)]

19. Xie, S.; Imani, M.; Dougherty, E.R.; Braga-Neto, U.M. Nonstationary linear discriminant analysis. In Proceedings of the IEEE 2017 51st Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, CA, USA, 29 October–1 November 2017; pp. 161–165, doi:10.1109/ACSSC.2017.8335158. [[CrossRef](#)]
20. Imani, M.; Ghoreishi, S.F.; Braga-Neto, U.M. Bayesian control of large MDPs with unknown dynamics in data-poor environments. In Proceedings of the Advances in Neural Information Processing Systems (NIPS), Montréal, QC, Canada, 3–8 December 2018, pp. 8146–8156.
21. Imani, M.; Ghoreishi, S.F.; Allaire, D.; Braga-Neto, U.M. MFBO-SSM: Multi-Fidelity Bayesian Optimization for Fast Inference in State-Space Models. In Proceedings of the AAAI Conference on Artificial Intelligence, Hilton Hawaiian Village, Honolulu, HI, USA, 27 January 27–1 February 2019; Volume 33, pp. 7858–7865, doi:10.1609/aaai.v33i01.33017858. [[CrossRef](#)]
22. Imani, M.; Dougherty, E.R.; Braga-Neto, U. Boolean Kalman filter and smoother under model uncertainty. *Automatica* **2020**, *111*, 108609, doi:10.1016/j.automatica.2019.108609. [[CrossRef](#)]
23. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149, doi:10.1109/TPAMI.2016.2577031. [[CrossRef](#)]
24. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; Volume 2016-Decem, pp. 779–788, doi:10.1109/CVPR.2016.91. [[CrossRef](#)]
25. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. In *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*; Springer International Publishing: Cham, Switzerland, 2016; Volume 9905 LNCS, pp. 21–37, doi:10.1007/978-3-319-46448-0_2. [[CrossRef](#)]
26. Chen, Y.; Zhu, K.; Zhu, L.; He, X.; Ghamisi, P.; Benediktsson, J.A. Automatic Design of Convolutional Neural Network for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 7048–7066, doi:10.1109/TGRS.2019.2910603. [[CrossRef](#)]
27. Liu, Q.; Feng, C.; Song, Z.; Louis, J.; Zhou, J. Deep Learning Model Comparison for Vision-Based Classification of Full/Empty-Load Trucks in Earthmoving Operations. *Appl. Sci.* **2019**, *9*, doi:10.3390/app9224871. [[CrossRef](#)]
28. Mousavi, A.; Baraniuk, R.G. Learning to invert: Signal recovery via Deep Convolutional Networks. In Proceedings of the 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), New Orleans, LA, USA, 5–9 March 2017; pp. 2272–2276, doi:10.1109/ICASSP.2017.7952561. [[CrossRef](#)]
29. Kulkarni, K.; Lohit, S.; Turaga, P.; Kerviche, R.; Ashok, A. ReconNet: Non-Iterative Reconstruction of Images from Compressively Sensed Measurements. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 449–458, doi:10.1109/CVPR.2016.55. [[CrossRef](#)]
30. Yao, H.; Dai, F.; Zhang, S.; Zhang, Y.; Tian, Q.; Xu, C. DR2-Net: Deep Residual Reconstruction Network for image compressive sensing. *Neurocomputing* **2019**, *359*, 483–493, doi:10.1016/j.neucom.2019.05.006. [[CrossRef](#)]
31. Ballé, J.; Laparra, V.; Simoncelli, E.P. End-to-end Optimized Image Compression. In Proceedings of the 5th International Conference on Learning Representations, ICLR 2017, Conference Track Proceedings, Toulon, France, 24–26 April 2017.
32. Wang, C.; Han, Y.; Wang, W. An End-to-End Deep Learning Image Compression Framework Based on Semantic Analysis. *Appl. Sci.* **2019**, *9*, 3580. [[CrossRef](#)]
33. Lyu, C.; Liu, Z.; Yu, L. Block-sparsity recovery via recurrent neural network. *Signal Process.* **2019**, *154*, 129–135. doi:10.1016/j.sigpro.2018.08.014. [[CrossRef](#)]
34. Toderici, G.; Vincent, D.; Johnston, N.; Jin Hwang, S.; Minnen, D.; Shor, J.; Covell, M. Full Resolution Image Compression with Recurrent Neural Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
35. Minnen, D.; Toderici, G.; Covell, M.; Chinen, T.; Johnston, N.; Shor, J.; Hwang, S.J.; Vincent, D.; Singh, S. Spatially adaptive image compression using a tiled deep network. In Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP), Beijing, China, 17–20 September 2017; pp. 2796–2800, doi:10.1109/ICIP.2017.8296792. [[CrossRef](#)]

36. Ma, S.; Zhang, X.; Jia, C.; Zhao, Z.; Wang, S.; Wang, S. Image and Video Compression with Neural Networks: A Review. *IEEE Trans. Circuits Syst. Video Technol.* **2019**, *8215*, 1, doi:10.1109/TCSVT.2019.2910119. [[CrossRef](#)]
37. Rippel, O.; Bourdev, L. Real-time adaptive image compression. In Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, Australia, 6–11 August 2017; Volume 6, pp. 4457–4473.
38. Jia, C.; Zhang, X.; Wang, S.; Wang, S.; Ma, S. Light Field Image Compression Using Generative Adversarial Network-Based View Synthesis. *IEEE J. Emerg. Sel. Top. Circuits Syst.* **2019**, *9*, 177–189, doi:10.1109/JETCAS.2018.2886642. [[CrossRef](#)]
39. Bjontegaard, G. Calculation of average PSNR differences between RD-curves. In Proceedings of the VCEG Meeting (ITU-T SG16 Q.6), Austin, TX, USA, 2–4 April 2001, pp. 2–4.
40. Agustsson, E.; Tschannen, M.; Mentzer, F.; Timofte, R.; Gool, L.V. Generative Adversarial Networks for Extreme Learned Image Compression. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Seoul, Korea, 27 October–2 November 2019.
41. He, K.; Zhang, X.; Ren, S.; Sun, J. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 1026–1034, doi:10.1109/ICCV.2015.123. [[CrossRef](#)]
42. Miklós, P. Image interpolation techniques. In Proceedings of the 2nd Siberian-Hungarian Joint Symposium On Intelligent Systems, Subotica, Serbia and Montenegro, 1–2 October 2004; pp. 1–6.
43. Ioffe, S.; Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6–11 July 2015; Volume 1, pp. 448–456.
44. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778, doi:10.1109/CVPR.2016.90. [[CrossRef](#)]
45. Maas, A.L.; Hannun, A.Y.; Ng, A.Y. Rectifier nonlinearities improve neural network acoustic models. In Proceedings of the ICML Workshop on Deep Learning for Audio, Speech and Language Processing, Atlanta, GA, USA, 16 June 2013.
46. Wang, Z.; Bovik, A.C. Mean squared error: Love it or leave it? A new look at Signal Fidelity Measures. *IEEE Signal Process. Mag.* **2009**, *26*, 98–117, doi:10.1109/MSP.2008.930649. [[CrossRef](#)]
47. Wang, Z.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Process.* **2004**, *13*, 600–612, doi:10.1109/TIP.2003.819861. [[CrossRef](#)] [[PubMed](#)]
48. Sampat, M.P.; Wang, Z.; Gupta, S.; Bovik, A.C.; Markey, M.K. Complex Wavelet Structural Similarity: A New Image Similarity Index. *IEEE Trans. Image Process.* **2009**, *18*, 2385–2401, doi:10.1109/TIP.2009.2025923. [[CrossRef](#)]
49. Mallat, S.G. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Trans. Pattern Anal. Mach. Intell.* **1989**, *11*, 674–693, doi:10.1109/34.192463. [[CrossRef](#)]
50. Kotu, V.; Deshpande, B. Classification. In *Data Science*; Elsevier: Amsterdam, The Netherlands, 2019; pp. 65–163, doi:10.1016/B978-0-12-814761-0.00004-6. [[CrossRef](#)]
51. Kingma, D.P.; Ba, J.L. Adam: A method for stochastic optimization. In Proceedings of the 3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings, San Diego, CA, USA, 7–9 May 2015.
52. Lin, T.Y.; Maire, M.; Belongie, S.; Bourdev, L.; Girshick, R.; Hays, J.; Perona, P.; Ramanan, D.; Zitnick, C.L.; Dollár, P. Microsoft COCO: Common Objects in Context. In *Computer Vision—ECCV 2014*; Springer International Publishing: Cham, Switzerland, 2014; Volume 8693 LNCS, pp. 740–755 doi:10.1007/978-3-319-10602-1_48. [[CrossRef](#)]
53. Paszke, A.; Chintala, S.; Chanan, G.; Lin, Z.; Gross, S.; Yang, E.; Antiga, L.; Devito, Z.; Lerer, A.; Desmaison, A. Automatic differentiation in PyTorch. In Proceedings of the 31st Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017.

