applied sciences

MDPI

Article

Deep Image Compression with Residual Learning

Weigui Li, Wenyu Sun, Yadong Zhao, Zhuqing Yuan and Yongpan Liu *

Department of Electronic Engineering, Tsinghua University, Beijing 100081, China;
lwg17@mails.tsinghua.edu.cn (W.L.); wy-sun16@mails.tsinghua.edu.cn (W.S.);
zyd17@mails.tsinghua.edu.cn (Y.Z.); yuanzhuqing@tsinghua.edu.cn (Z.Y.)
* Correspondence: ypliu@tsinghua.edu.cn

Received: 7 May 2020; Accepted: 2 June 2020; Published: 10 June 2020

Abstract: An end-to-end image compression framework based on deep residual learning is proposed. Three levels of residual learning are adopted to improve the compression quality: (1) the ResNet structure; (2) the deep channel residual learning for quantization; and (3) the global residual learning in full resolution. Residual distribution is commonly a single Gaussian distribution, and relatively easy to be learned by the neural network. Furthermore, an attention model is combined in the proposed framework to compress regions of an image with different bits adaptively. Across the experimental results on Kodak PhotoCD test set, the proposed approach outperforms JPEG and JPEG2000 by PSNR and MS-SSIM at low BPP (bit per pixel). Furthermore, it can produce much better visual quality. Compared to the state-of-the-art deep learning-based codecs, the proposed approach also achieves competitive performance.

Keywords: deep learning; image compression; residual learning

1. Introduction

Image compression is a fundamental and well-studied problem in the data compression field. Typical conventional compression algorithms such as JPEG [1] and JPEG2000 [2] are based on transform coding theory [3]. JPEG adopts the discrete cosine transform (DCT) while JPEG2000 adopts the discrete wavelet transform (DWT) to convert an original image into a latent compression representation space.

Recently, lossy image compression frameworks based on deep learning (DL) have raised interest in both deep learning and image processing communities [4–10]. Although these approaches are competitive with the existing modern engineered codecs such as JPEG [1], JPEG2000 [2], WebP [11], and BPG [12], several issues and challenges still need to be addressed. Firstly, existing learning methods commonly compress the full resolution image at one-shot regardless of the difference between high- and low-frequency information, which brings out some artifacts and block effects. Secondly, data compression is mainly reflected in the quantization stage, but it may introduce too many unnecessary errors. Based on previous works, a residual learning framework is proposed by introducing some novel technologies to properly improve the compression quality.

In Section 1.1, DL-based image compression related works are reviewed and discussed. In Section 1.2, deep channel residual learning is introduced. In Section 1.3, global residual learning in full resolution is presented. In Section 1.4, the main contributions of this paper are summarized.

1.1. DL-Based Image Compression and Related Works

Generally, a typical end-to-end lossy image compression framework can be formulated as jointly optimizing all modules: an encoder E_ϕ, a quantizer Q, a decoder D_θ, and some other rate estimation and rate-distortion control module R, where ϕ and θ are the trained weights.

$$E_\phi : \mathbb{R}^N \to \mathbb{R}^M \qquad D_\theta : \mathbb{R}^M \to \mathbb{R}^N \qquad Q : \mathbb{Z}^M \to [0,1] \tag{1}$$

A vector of image intensities $x \in \mathbb{R}^N$ is mapped to a latent presentation space $y \in \mathbb{R}^M$ via the encoder E_ϕ, $y = E_\phi(x)$. y is quantized into a discrete-valued vector $\hat{q} \in Z^M$ via the quantizer Q. The entropy rate of the vector $\hat{q}$ is estimated and minimized via R. The reconstructed image can be represented as $\tilde{x} = D_\theta(\tilde{q})$. Distortion is often assessed by the distance between x and $\tilde{x}$, $D(x, \tilde{x})$, namely PSNR (Peak Signal to Noise Ratio), or the perceptual measure such as MS-SSIM (Multi-scale Structural Similarity) [13]. Rate estimation is often assessed by the minimal entropy of the discrete bit streams. Therefore, image compression is typically formulated as a distortion and rate optimization problem. The goal is o optimize the trade-off between using a minimal number of bits and having minimal distortion.

$$Optimization \quad object: \quad \underbrace{D(x,\tilde{x})}_{Distortion} - \lambda \cdot \underbrace{log_2 Q(E_\phi(x))}_{Number\ of\ bits} \tag{2}$$

Existing DL-based image compression frameworks are mainly classified into the auto-encoder and recurrent neural network (RNN). In the case of auto-encoder, recent works have focused on variational Bayes and autoregressive context methods to develop accurate entropy estimation models by optimizing the R-D curve and introducing hyper prior parameters [6,8]. The prior probability of compression representation is modeled by a Gaussian or Laplacian distribution by an underlying auto-encoder. Since the Lagrange multiplier λ is introduced as a hyper-parameter for a desired trade-off between the rate and distortion, it is not efficient and different models must be trained for different compression rates.

Due to a large fraction of Internet traffic driven by requests from mobile devices, a progressive algorithm is more popular. Recurrent neural networks (RNNs)-based frameworks only need to be trained once to produce a progressive and variable-rate model, regardless of the desired compression rate [14,15]. However, the RNN-based methods require multiple iterations to obtain high-quality images. Besides, Agustsson et al. [4] proposed a generative adversarial network to obtain visually pleasing results at low bit rates where previous methods fail and show strong artifacts. Although the PSNR and MS-SSIM index shows no advantage, the reconstructed images seem better than others at extremely low bit rates.

Combined with the advantages of both, the proposed framework adopts the recurrent architecture with rate control to learn residual information.

1.2. Deep Channel Residual Learning

Since quantization is necessary for compression, it also introduces a lot of errors. Assuming that the shape of the quantized bits is $n \times h \times w$, the quantization errors exist in these coded bits. Motivated by denoising algorithm in traditional signal processing, a variation of $\Delta\Sigma$ modulation similar to residual learning is introduced. Utilizing the residual information between the adjacent channel can eliminate errors effectively. Given the input is x, the quantizer is *sign* function, and the quantized bits is $\hat{q}$. The residual is r, i.e. the difference between current channel y_i and previous quantized channel $\hat{q}_i$. The reconstruction z is accumulated by residual r. The errors existed between channels can be eliminated in the process of calculating the residual and accumulation of the information. The specific process of channel residual learning is depicted as Algorithm 1.

Algorithm 1: Deep channel residual learning.

```
Input: Up-sampled representations $y_1, y_2, ..., y_n$
Output: Quantized bits $\hat{q}_1, \hat{q}_2, ..., \hat{q}_n$
$r \Leftarrow 0, z \Leftarrow y_1$
for i = 1,2,...,n do
$z \Leftarrow z + r$
$\hat{q}_i \Leftarrow sign(z)$
$r \Leftarrow y_i - \hat{q}_i$
end for
```

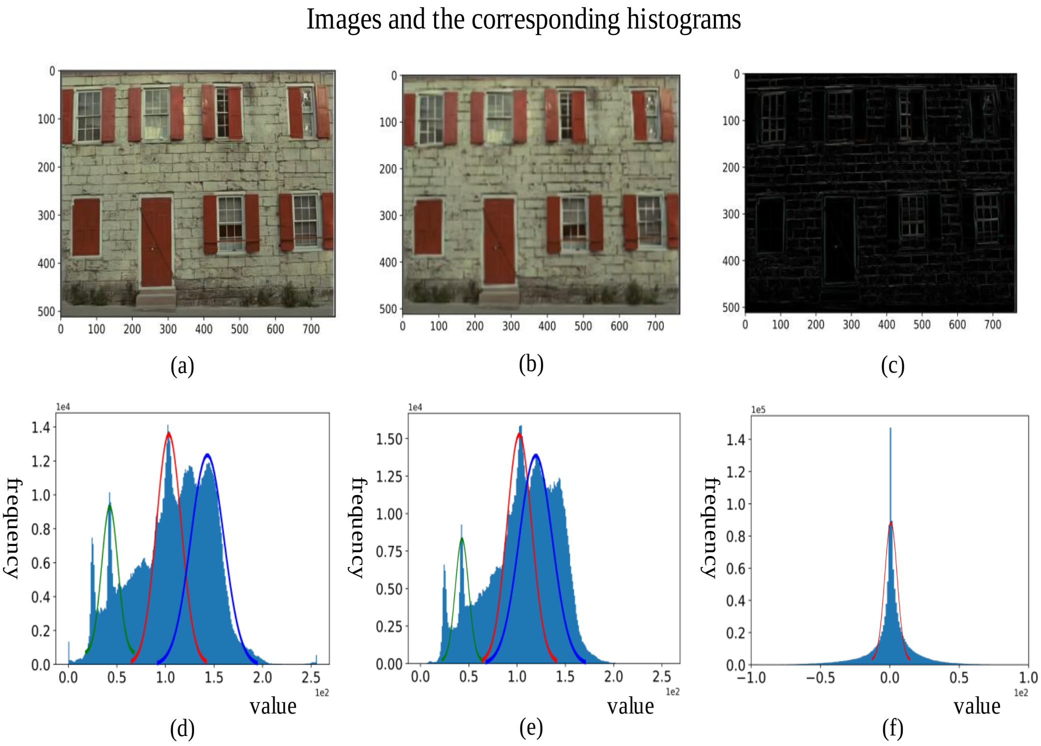
1.3. Global Residual Learning

An image is made up of low- and high-frequency components. In traditional image compression methods, most of the high-frequency information is discarded and the low-frequency part is well preserved. DL-based methods are trying to learn more representative features, both high- and low-frequency, while it requires fewer bits. Therefore, how to effectively learn high-frequency residual information is crucial.

1.3.1. Global Residual

In a probabilistic generative model such as variational inference, the goal is to approximate the true posterior $p_{\tilde{y}|x}(\tilde{y}|x)$ with a parametric variational density $q(\tilde{y}|x)$ by minimizing the expectation of the Kullback–Leibler (KL) divergence between image distribution p_x and $q(\tilde{y}|x)$. In a generative adversarial model, the goal is to directly generate a fake image to deceive the discriminator. Assuming that the original distribution is p_x, and the reconstruction distribution is $p_{\tilde{x}}$, the residual distribution is $p_{\Delta x} = p_{(x-\tilde{x})}$. The above generative model always tries to generate the original distribution p_x.

Figure 1 shows the original image, reconstruction image, residual image, and the corresponding histograms, respectively. Comparing the histogram of the original to that of the reconstruction, the residual's histogram is similar to a single Gaussian distribution that makes the network easier to learn it.



(a) (b) (c)

(d) (e) (f)

Figure 1. Images and the corresponding histograms: (**a**) original; (**b**) reconstruction; (**c**) residual; (**d**) original histogram; (**e**) reconstruction histogram; and (**f**) residual histogram.

For this reason, another specific learning branch is designed to learn the residual information so that the network can accumulate more details for reconstruction.

1.3.2. Side Information

Few additional coded messages can improve the compression efficiency and quality. For example, block partition messages are often coded and sent to the decoder as side information in traditional JPEG [1], JPEG2000 [2], and H.265 [16] codecs. Ballé et al. [6] and Lee et al. [17] sent standard deviations $\hat{\sigma}$ with a hyperprior as side information to the latent representations. Agustsson et al. [4] sent additional image semantic information to the generator for better visual quality. Here, compressing the residual with another few bits as side information is adopted to reconstruct the original image.

1.4. Contributions and Paper Organizations

In this paper, the residual learning method is firstly proposed to individually compress the low- and high-frequency information. It reduces not only the difficulty of training model but also the complexity of generating full resolution image distribution. Secondly, it specifically explains and demonstrates that the essence of DL-based image compression is the process of residual learning. Moreover, this paper fully taps the potential of three-level residual learning and improves some existing technical methods in image compression.

The rest of this paper is organized as follows. In Section 2, the proposed framework and the details of each module are illustrated. Section 3 describes some experimental results to demonstrate the effectiveness of all modules. Comparison results with traditional and DL-based codecs are shown. Furthermore, ablation studies are also presented. Section 4 concludes this paper.

2. Proposed Framework

2.1. Overview of the Proposed Framework

The proposed image compression framework is composed of two parts: one is for the main information compression, and the other is for the residual information compression. They both contain convolutional encoder, channel residual quantizer, and convolutional decoder. Figure 2 shows the proposed framework.

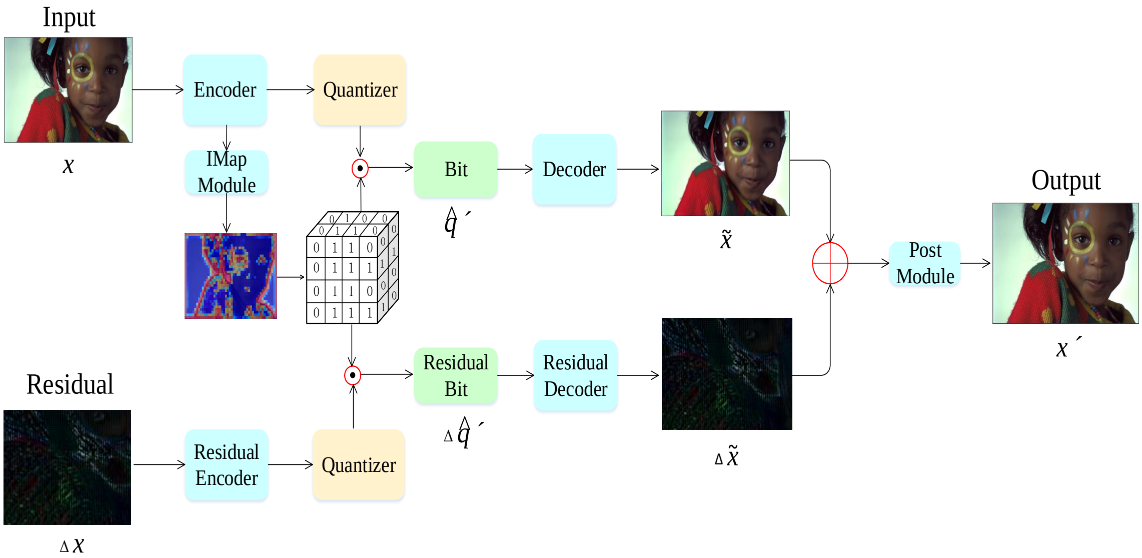


Figure 2. Proposed image compression framework. Note that, for better visualization, the salient region of the residual image was cropped.

From the encoder side, a vector of image intensities x is transformed into latent representation space y, quantized to $\hat{q}$, multiplied by importance map b to $\hat{q}'$, and then sent to the decoder for reconstruction $\tilde{x}$. Then, the residual $\Delta x(\Delta x = x - \tilde{x})$ is successively transformed to Δy by residual encoder, to $\Delta\hat{q}$ by quantizer, and to $\Delta\hat{q}'$ by multiplying the same importance map b. Finally, residual decoder transforms $\Delta\hat{q}'$ to $\Delta\tilde{x}$ for residual reconstruction. The reconstructed image x' is the accumulated outputs of the decoder and residual decoder, x′ = $\tilde{x}$ + $\Delta\tilde{x}$. The coded bit streams are consisted of all quantized bits, q = $\hat{q}'$ + $\Delta\hat{q}'$.

For the encoder, the input image is scaled to $\frac{1}{2^M} \cdot \frac{1}{2^M}$ through M residual blocks with convolutional stride 2 instead of MaxPooling layers. Simple and effective group normalization (GN) was adopted instead of generalized divisive normalization (GDN) mentioned in [6,18]. Before quantization, the channel number is up-sampled to N_{out}. On the decoder side, another channel down-sampling and pixel up-scaling operations are taken to reconstruct the image.

2.2. Residual Structure

In the proposed compression framework, both the encoder and decoder are fully convolutional structure regardless of the input image size. The residual blocks are the basic components of the network as shown in Figure 3, which is derived from ResNet architecture firstly proposed for image recognition [19].

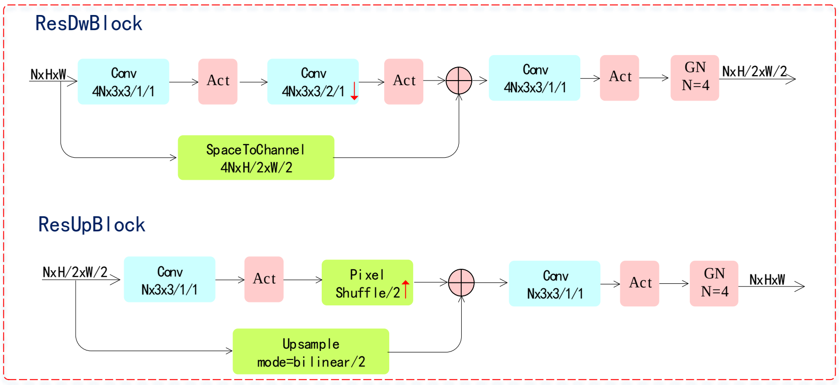


Figure 3. ResDwBlock and ResUpBlock.

2.3. Encoder and Decoder

The encoder consists of four residual blocks with stride 2 for down-sampling. Batch normalization was found to have no beneficial effect and Ballé *et al.* [18] proved GDN was a good choice. However, group normalization (GN) [20] is simple and also a better choice from the experiments. It is more suitable for local divisive normalization and training large models which require small batches constrained by memory consumption. Moreover, the shortcut structure is introduced to feature fusion and extraction, which transmits more useful information to the following concatenate layer. The decoder also consists of four residual blocks with PixelShuffle layers for up-sampling. The activation function is leakyReLU. The details of the architecture are shown in Figure 4.

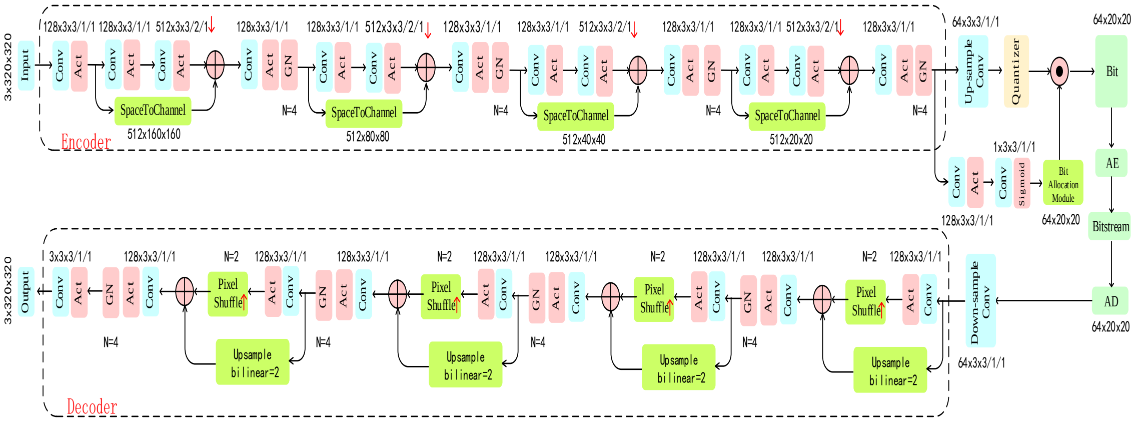


Figure 4. Illustration of the encoder and decoder. The blue box represents residual blocks for down-sampling or up-sampling, convolutional parameters are denoted as filter × kernel height × kernel width/stride/padding, the pink box represents group normalization (GN), activation function (Act), the yellow box is quantizer, the green box represents special operation module, and AE and AD represent arithmetic encoder and arithmetic decoder, respectively.

2.4. Channel Residual Quantization

Quantization is the main part of compression task, which not only compresses data but also introduces errors and noises. Binary representation is an important mean of quantization. A binary bottleneck helps force the network to discard more redundant bit patterns comparing to standard round function or floating-point representation. This paper follows Raiko et al. [21] and Toderici et al. [14] by quantizing the input value $x \in (-1, 1)$ to the set $\{-1, 1\}$ for each value. Assuming that $b(x) \in \{-1, 1\}$ and $x \in (-1, 1)$, it can be defined as:

$$b(x) = x + \varepsilon \qquad \in \{-1, 1\} \tag{3}$$

$$\varepsilon \sim \begin{cases} -x + 1 & with \quad probability \quad \frac{1+x}{2} \\ -x - 1 & with \quad probability \quad \frac{1-x}{2} \end{cases} \tag{4}$$

where ε denotes the quantization noise. For the backward propagation, the derivative of the expectation $\mathbb{E}[b(x)] = x$, for $x \in (-1, 1)$, is used. Therefore, the gradient back propagated through $b(x)$ is 1. Once the networks are trained, $b(x)$ is replaced by b^{inf} defined as:

$$b^{inf}(x) = \begin{cases} -1 & if \quad x < 0 \\ +1 & \quad otherwise \end{cases} \tag{5}$$

As analyzed in Section 1.2, deep channel residual learning method can be used to eliminate the errors between channels to some extent. Therefore, combining the two technologies, a denoising quantization method is employed.

2.5. Improved Importance Map Method for Rate Control

Importance map method for rate control was firstly proposed by Li et al. [7]. Instead of rounding and entropy, they defined rate loss on importance map and adopted a simple binarization for quantization. This method for rate control is content-weighted and benefit to improve visual quality. Here, the part of the architecture in [7] and some novel technologies (e.g., binarization with probability depicted in Section 2.4 and smooth bit allocation with sigmoid function) are adopted. The process is shown in Figure 4.

Intermediate feature maps are utilized to yield content-weighted importance map $p(x)$. The importance map has only one channel, with the same size as the encoder and the range of (0,1). The value of the map determines the amount of bit allocation and the compression level. A mask $b(x)$ from $p(x)$ with the same size as $\hat{q}$ is generated. The final quantized bit $\hat{q}' = b(x) \odot \hat{q}$.

Assuming that $1 \times h \times w$ is the size of $p(x)$, n is the up-sampling filter, so the size of the mask is $n \times h \times w$ and the size of the $\hat{q}$ is $n \times h \times w$. Here, a recurrent bit allocation method was adopted. Given the bit allocation step s, the bit allocation amount in every step and channel direction is $m = \frac{n}{s}$. Assuming that a value is $p_{ij}(x)$ in $p(x)$ and the $b(p_{ij})$ is the mask value in this step, the allocation method is defined as follows:

$$b(p_{ij})(x) = \begin{cases} 1 & if \quad (p_{ij} - \frac{i}{s}) > 0, \quad i = 0, 1, ..., s \\ 0 & \quad otherwise \end{cases} \tag{6}$$

The calculation formula is:

$$b(p_{ij}) = \frac{1}{2} \times (sign(2 \times sigmoid(k \times (p_{ij} - \frac{i}{s})) - 1) + 1) \tag{7}$$

where k is a constant, and $i \in \{0, 1, ..., s\}$, the size of $b(p_{ij})$ is $m \times h \times w$. When the sth step is completed, a mask $b(x)$ with the size of $n \times h \times w$ is obtained.

The quantization in this stage follows the method depicted in Section 2.4.

2.6. Training Strategy

2.6.1. Loss Function

Generally, jointly optimizing the rate loss and distortion loss is necessary. A trade-off parameter λ is used to balance compression rate and distortion. Moreover, another residual distortion loss $D(\Delta x, \Delta\tilde{x})$ is applied. Therefore, the objective function is defined as follows:

$$\mathcal{L} = \sum \left\{ \mathcal{L}_{D_1}(x, \tilde{x}) + \gamma \mathcal{L}_{D_2}(\Delta x, \Delta\tilde{x}) + \lambda \mathcal{L}_R(x) \right\} \tag{8}$$

Distortion Loss. As mentioned above, distortion loss is to evaluate the difference between the original image and reconstruction image. Mean square error (MSE), mean absolute error (MAE), or perceptual space loss (MS-SSIM) is used to define it. In this paper, the distortion is assessed by MSE loss. Besides, another residual loss is added to obtain better visual effect. The total distortion is defined as follows:

$$\mathcal{L}_D = \sum \left\{ \|x - \tilde{x}\|^2 + \gamma \, \|\Delta x - \Delta\tilde{x}\|^2 \right\} \tag{9}$$

Rate Loss. Importance map is a substitution of entropy estimation. Proper optimization of the bit allocation through the importance map is a good choice to minimize the entropy of the coded bits. Compared to the monotonous region, the region of rich texture gets more coded bits. The optimal goal is to encode the most abundant information with the fewest bits. Therefore, the rate loss can be defined as follows:

$$\mathcal{L}_R = \frac{1}{M} \sum_{i,j}^{m} p_{ij}(x) \tag{10}$$

2.6.2. Optimization Function and Training Details

ADAM optimizer [22] with an initial learning rate $2{\times}10^{-4}$ is employed. Multi-step learning rate schedule is applied and the decay factor is 0.5. The activation function is leakyReLU. The training hyper-parameters, such as the iteration of residual learning r, the bit allocation step s, the residual blocks' channel n_f, and the up-sampling filter N_{out}, are adjustable. In this paper, r = 1, 3; s = 8; n_f = 32, 64, 128; and N_{out} = 32, 64, 128, are set, respectively. In the training process, first the focus is on optimization on the distortion loss, and then it transfers to optimizing the rate loss. Note that increasing the training parameter value can improve the index, but it would also increase the cost and difficulty of training.

3. Experiments

3.1. Experimental Setup

The results reported in this paper were trained on the dataset from Challenge on Learned Image Compression (CLIC). To improve performance, extra images from Flickr2K and DIV2K datasets were added to the training set at the cost of longer training time. The high-quality images were randomly cropped to 320 × 320 pixels and saved as lossless PNGs, with about 100,000 patches in total. The CLIC validation dataset and Kodak PhotoCD image dataset were used to evaluate our models. The compression rate is defined by the bits per pixel (BPP) as follows:

$$BPP = \frac{N_{out} + \Delta N_{out}}{2^M \cdot 2^M} \cdot \mathbb{E}(b(p_{ij})) \tag{11}$$

where N_{out} and ΔN_{out} are the up-sampling channels, M is the down-sampling number, and $\mathbb{E}(b(p_{ij}))$ is the mean BPP ratio in importance map. The image distortion was evaluated by PSNR and MS-SSIM. Some experiments were finished with the parameters λ = $5{\times}10^{-3}$, 1 $\times 10^{-2}$, $5{\times}10^{-2}$, and N_{out} = 32, 64, 128, $M = 4$, corresponding to BPP approximately in the range of (0.1,0.5).

In the following, the results of both quantitative metrics and visual quality evaluation are presented, and some ablation studies are shown.

3.2. Experimental Results

3.2.1. Quantitative Evaluation

All trained models were evaluated on the publicly available Kodak dataset (Eastman Kodak, 1993) [23], an uncompressed set of images commonly used to evaluate image compression methods. Image distortion were optimized for MSE and summarized rate-distortion curves are shown in Figure 5. For quantitative evaluation, the test results with JPEG [1], JPEG2000 [2], and the DL-based previous methods such as Theis et al. (2017) [10], Li et al. (2017) [7], and Ballé et al. (2018) [6] were compared.

In this paper, the distortion is only assessed by mean square error (MSE) loss. Thus, the results shown in Figure 5 are under MSE optimization.Although our approach cannot achieve the state-of-the-art results, the proposed approach outperforms JPEG, JPEG2000, and Theis et al. [10] in PSNR and MS-SSIM index at low BPP. In this paper, the low bit range is less than 0.5 BPP. The corresponding compression ratio is more than 48 times. When the BPP is set 0.1, the corresponding compression ratio is 240 times. It is worth mentioning that the proposed framework performs better at low bits. At low bits, the compression rate is higher, and more residual information is discarded. Thus, the specific residual learning network is more suitable to learn representative residual features.

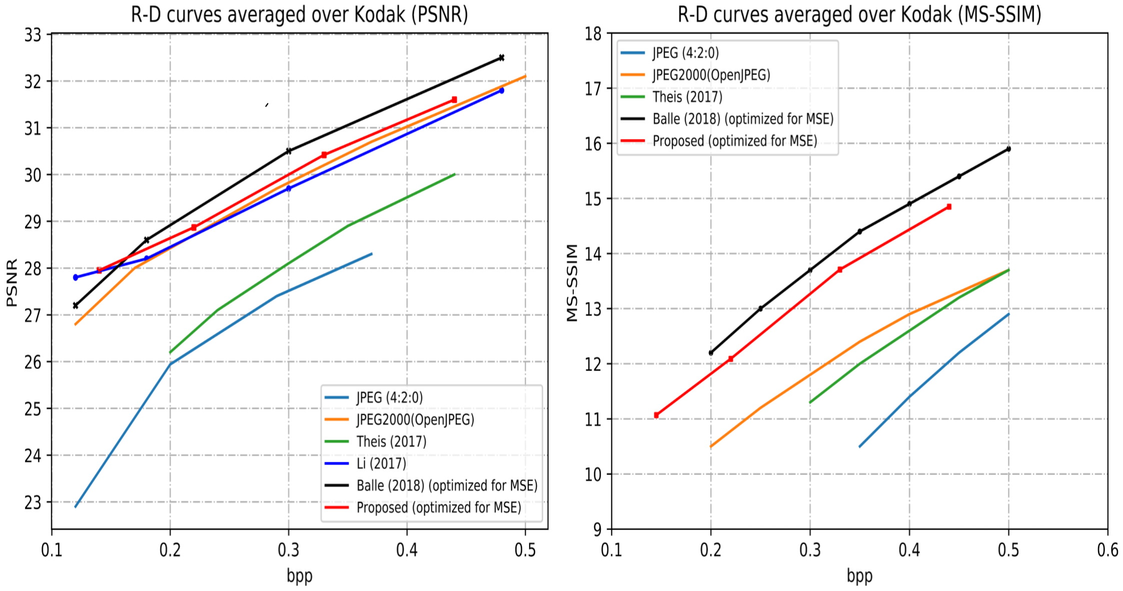


Figure 5. Rate–distortion curves of the proposed method and competitive approaches over the Kodak dataset. The measures are the PSNR and the MS-SSIM value as a function of BPP. Note that the PSNR's formula is $10\times log_{10}\frac{255^2}{MSE}$, and MS-SSIM values are converted into decibels for better visualization $(-10 \times log_{10}(1 - MSSSIM))$.

3.2.2. Visual Quality Evaluation

Furthermore, visual quality evaluation is arranged to demonstrate the effectiveness of the deep residual learning algorithm by comparing it with JPEG, JPEG2000, and Ballé (2018). Figures 6–8 clearly show that artifacts, blurring, and blocking are obvious the in traditional image compression algorithms, e.g., JPEG and JPEG2000. Although Ballé's (2018) PSNR is higher than the proposed model, the visual quality is not good enough. The proposed framework is good at eliminating blocking effect and smoothing the blurs.

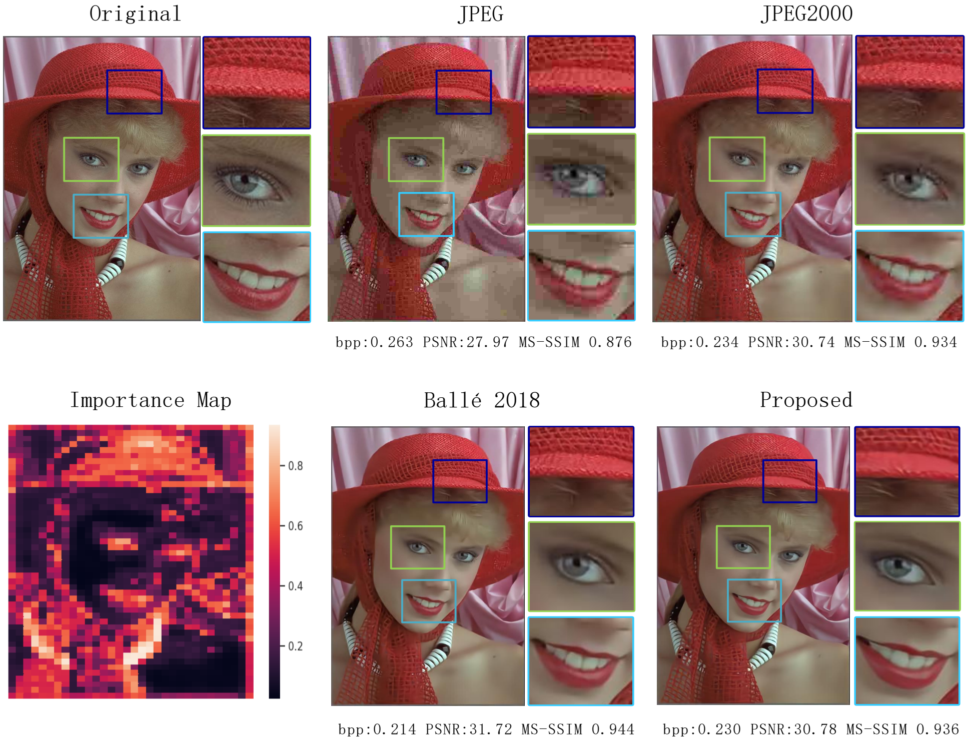

Original
JPEG
JPEG2000
bpp:0.263 PSNR:27.97 MS-SSIM 0.876
bpp:0.234 PSNR:30.74 MS-SSIM 0.934
Importance Map
0.8
0.6
0.4
0.2
Ballé 2018
Proposed
bpp:0.214 PSNR:31.72 MS-SSIM 0.944
bpp:0.230 PSNR:30.78 MS-SSIM 0.936

Figure 6. Visual quality evaluation of the specific image named kodim04 in Kodak dataset with different approaches.

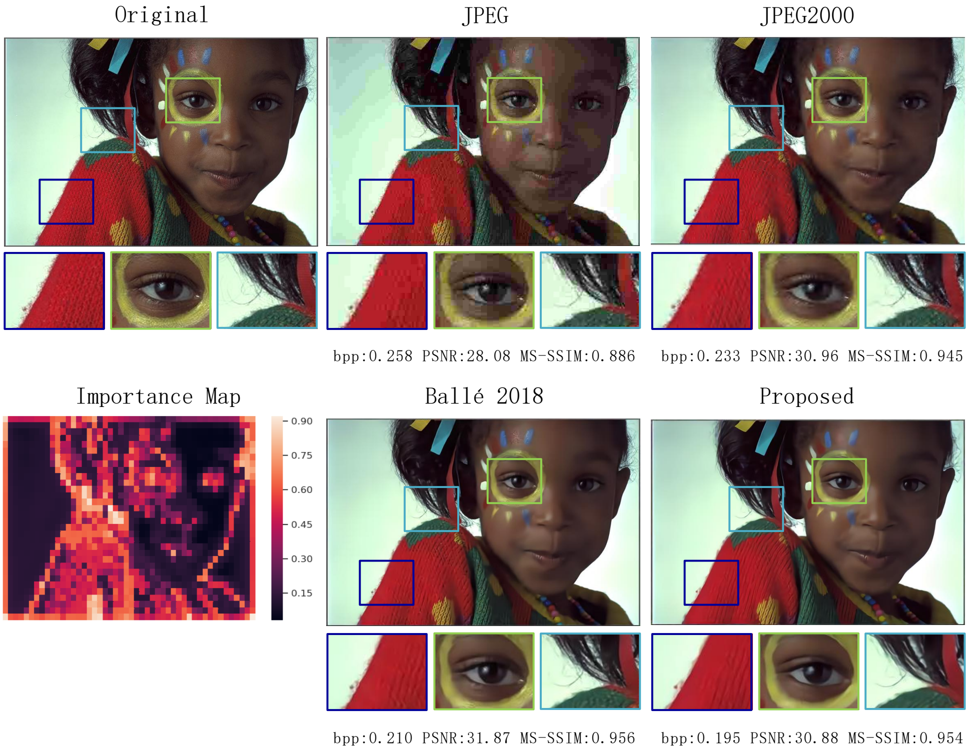

Original
JPEG
JPEG2000
bpp:0.258 PSNR:28.08 MS-SSIM:0.886
bpp:0.233 PSNR:30.96 MS-SSIM:0.945
Importance Map
0.90
0.75
0.60
0.45
0.30
0.15
Ballé 2018
Proposed
bpp:0.210 PSNR:31.87 MS-SSIM:0.956
bpp:0.195 PSNR:30.88 MS-SSIM:0.954

Figure 7. Visual quality evaluation of the specific image named kodim15 in Kodak dataset with different approaches.

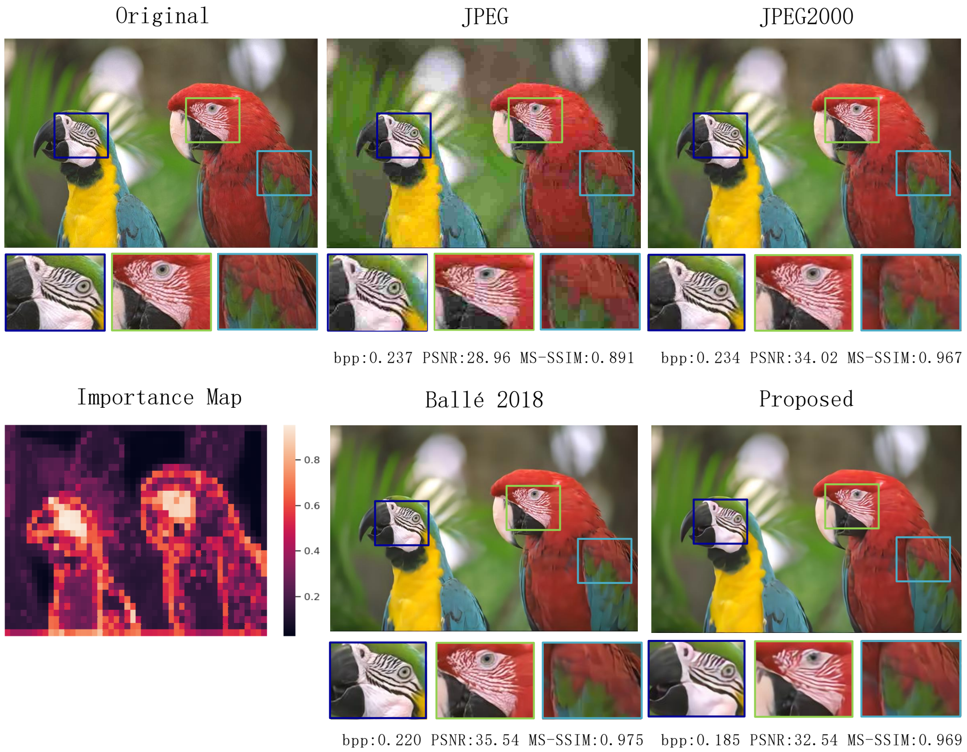

Original
JPEG
JPEG2000
bpp:0.237 PSNR:28.96 MS-SSIM:0.891
bpp:0.234 PSNR:34.02 MS-SSIM:0.967
Importance Map
Ballé 2018
Proposed
0.8
0.6
0.4
0.2
bpp:0.220 PSNR:35.54 MS-SSIM:0.975
bpp:0.185 PSNR:32.54 MS-SSIM:0.969

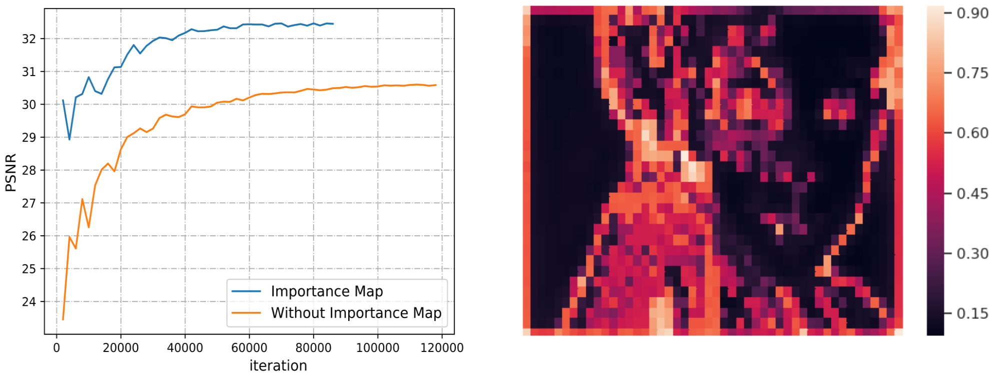
Figure 8. Visual quality evaluation of the specific image named kodim23 in Kodak dataset with different approaches.

3.3. *Ablation Studies*

To train a better image compression model, many ablation experiments were implemented. The results of ablation study are shown below, such as importance map for rate control, channel residual learning, global residual learning, etc.

3.3.1. Importance Map

To assess the role of importance map, the training process of the baseline model without importance map was compared to the model with importance map at about 0.47 BPP. For the sake of fairness, the sizes of the two models were the same approximately. The details during training phase are shown in Figure 9a. The importance map is used for rate estimation, and improves the performance by about 2 dB. Moreover, the importance map of the specific image named kodim15 in Kodak dataset is shown in Figure 9b. The figure exhibits that the red region represents the value $p_{ij}(x)$ is large, where more bits are allocated, such as the regions with rich texture and edge.



(a) (b)

Figure 9. (**a**) PSNR performance during the training phase and the process of residual learning at about 0.47 BPP; amd (**b**) importance map of image kodim15 in Kodak dataset.

3.3.2. Channel Residual Learning

As with the above experiments, a baseline model without channel residual learning and another model with channel residual learning were implemented at about 0.47 BPP on CLIC test dataset. As shown in Figure 10, the PSNR is also improved about 0.4 dB. Therefore, channel direction residual learning may eliminate the noise, as described in Section 1.2.

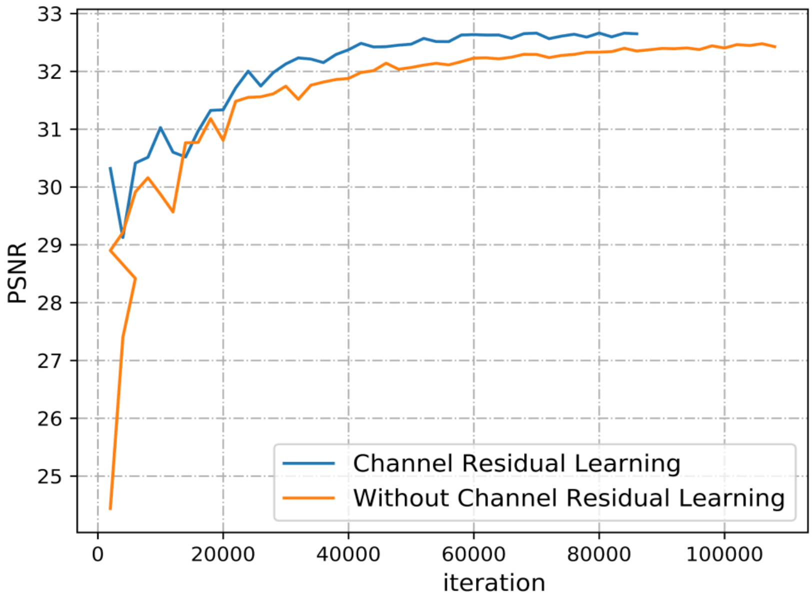


Figure 10. Results of ablation experiments about channel residual quantization.

3.3.3. Global Residual Learning

In Figure 5, the performance of the proposed framework is better than the method proposed by Li et al. [7] in PSNR. The most improved performance is the result of global residual learning. The specific residual learning branch is designed to encode the residual information. As the side information, it is added to the quantized bits. The details of the global residual learning process are shown in Figure 11.

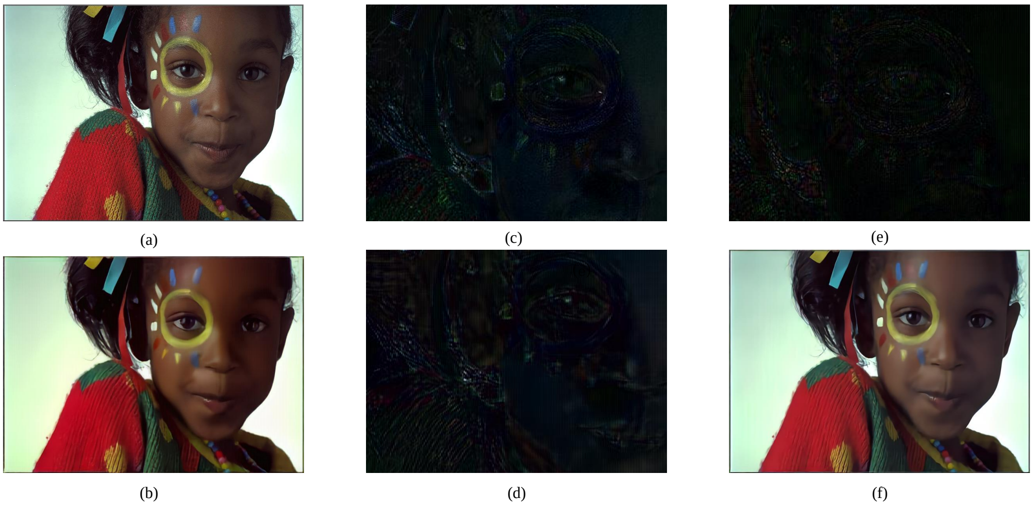


Figure 11. Process of global residual learning: (**a**) original x; (**b**) main information $\tilde{x}$; (**c**) residual Δx; (**d**) reconstructed residual $\Delta\tilde{x}$; (**e**) remained residual $\Delta x'$; and (**f**) reconstruction $x' = \tilde{x} + \Delta\tilde{x}$. Note that for better visualization, salient region of the residual image was cropped.

3.3.4. Other Experiments

Increasing the iteration of residual learning can also improve the performance at the cost of training time and training difficulty. Another model with residual block channels n_f = 128, up-sampling channels N_{out} = 64, and iteration of residual learning r = 3 was trained. Although the PSNR improved by about 0.5 dB, the model size increased by about two times.

To further improve performance and eliminate artifacts and blurs, a post-processing module similar to [24] is adopted to fine-tune on the trained model. Because the result of residual learning is added to the reconstruction image in this paper, the post-processing module can better integrate the information of both.

4. Conclusions

In this paper, a deep residual learning based image compression framework is proposed. Learning the residual distribution is easier compared to learning the full resolution information through the analysis of residual distribution. The high- and the -frequency components are separated to compress individually. At the same time, improved importance map methods are introduced to realize better bit allocation. Experimental results show that residual learning mechanism can improve the compression performance by focusing on key residual information. In the future, the entropy module will be introduced for rate estimation comparison and training models with higher BPP.

Author Contributions: Conceptualization, W.L., W.S., and Z.Y.; methodology, W.L., W.S., and Y.Z.; validation, W.L.; writing, W.L.; review and editing, W.L. and Y.Z.; and supervision, Y.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded in part by the National Key R&D Program under Grant 2017YFB1400900, and in part by NSFC under Grant 61934005.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Wallace, G. The JPEG still picture compression standard. *Commun. ACM* **1991**, *34*, 30–44. [CrossRef]
2. Rabbani, M. JPEG2000: Image Compression Fundamentals, Standards and Practice. *J. Electron. Imaging* **2002**, *11*. [CrossRef]
3. Goyal, V.K. Theoretical foundations of transform coding. *IEEE Signal Process. Mag.* **2001**, *18*, 9–21. [CrossRef]

4. Agustsson, E.; Tschannen, M.; Mentzer, F.; Timofte, R.; Gool, L.V. Generative Adversarial Networks for Extreme Learned Image Compression. *arXiv* **2018**, arXiv:1804.02958.
5. Choi, Y.; El-Khamy, M.; Lee, J. Variable Rate Deep Image Compression With a Conditional Autoencoder. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Seoul, Korea, October 2019.
6. Ballé, J.; Minnen, D.; Singh, S.; Hwang, S.J.; Johnston, N. Variational image compression with a scale hyperprior. *arXiv* **2018**, arXiv:1802.01436.
7. Li, M.; Zuo, W.; Gu, S.; Zhao, D.; Zhang, D. Learning Convolutional Networks for Content-weighted Compression. *arXiv* **2017**, arXiv:1703.10553v2.
8. Mentzer, F.; Agustsson, E.; Tschannen, M.; Timofte, R.; Van Gool, L. Conditional Probability Models for Deep Image Compression. *arXiv* **2018**, arXiv:1801.04260.
9. Rippel, O.; Bourdev, L. Real-Time Adaptive Image Compression. *arXiv* **2017**, arXiv:1705.05823.
10. Theis, L.; Shi, W.; Cunningham, A.; Huszár, F. Lossy Image Compression with Compressive Autoencoders. *arXiv* **2017**, arXiv:1703.00395.
11. Web. WebP Image Format. Available online: https://developers.google.com/speed/webp (accessed on 9 June 2020).
12. Bellard, F. BPG Image Format. Available online: https://bellard.org/bpg/ (accessed on 9 June 2020).
13. Wang, Z.; Simoncelli, E.P.; Bovik, A.C. Multiscale structural similarity for image quality assessment. In Proceedings of the Thrity-Seventh Asilomar Conference on Signals, Systems Computers, Pacific Grove, CA, USA, 9–12 November 2003; Volume 2, pp. 1398–1402. [CrossRef]
14. Toderici, G.; O'Malley, S.M.; Hwang, S.J.; Vincent, D.; Minnen, D.; Baluja, S.; Covell, M.; Sukthankar, R. Variable Rate Image Compression with Recurrent Neural Networks. *arXiv* **2015**, arXiv:1511.06085.
15. Johnston, N.; Vincent, D.; Minnen, D.; Covell, M.; Singh, S.; Chinen, T.T.; Hwang, S.J.; Shor, J.; Toderici, G. Improved Lossy Image Compression with Priming and Spatially Adaptive Bit Rates for Recurrent Networks. *arXiv* **2017**, arXiv:1703.10114.
16. Sullivan, G.J.; Ohm, J.; Han, W.; Wiegand, T. Overview of the High Efficiency Video Coding (HEVC) Standard. *IEEE Trans. Circuits Syst. Video Technol.* **2012**, *22*, 1649–1668. [CrossRef]
17. Lee, J.; Cho, S.; Beack, S.K. Context-adaptive Entropy Model for End-to-end Optimized Image Compression. *arXiv* **2018**, arXiv:1809.10452, .
18. Ballé, J.; Laparra, V.; Simoncelli, E.P. End-to-end Optimized Image Compression. *arXiv* **2016**, arXiv:1611.01704.
19. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. *arXiv* **2015**, arXiv:1512.03385.
20. Wu, Y.; He, K. Group Normalization. *arXiv* **2018**, arXiv:1803.08494.
21. Raiko, T.; Berglund, M.; Alain, G.; Dinh, L. Techniques for Learning Binary Stochastic Feedforward Neural Networks. *arXiv* **2014**, arXiv:1406.2989.
22. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. *arXiv* **2014**, arXiv:1412.6980.
23. Eastman Kodak. Kodak Lossless True Color Image Suite (Photocd pcd0992), 1993. Available online: http://r0k.us/graphics/kodak/ (accessed on 9 June 2020).
24. Zhou, L.; Cai, C.; Gao, Y.; Su, S.; Wu, J. Variational Autoencoder for Low Bit-rate Image Compression. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, Austin, TX, USA, June 2018.