Neural Image Compression via Attentional Multi-scale Back Projection and Frequency Decomposition

Ge Gao[1], Pei You[1], Rong Pan[1], Shunyuan Han[1], Yuanyuan Zhang[1], Yuchao Dai[2], Hojae Lee[1]
[1]Samsung R&D Institute China Xi'an, China [2]Northwestern Polytechnical University, Xi'an, China
[1]{ge1.gao, pei.you, rong.pan, shuny.han, yuan2.zhang, hojae72.lee}@samsung.com
[2]daiyuchao@nwpu.edu.cn

Abstract

In recent years, neural image compression emerges as a rapidly developing topic in computer vision, where the state-of-the-art approaches now exhibit superior compression performance than their conventional counterparts. Despite the great progress, current methods still have limitations in preserving fine spatial details for optimal reconstruction, especially at low compression rates. We make three contributions in tackling this issue. First, we develop a novel back projection method with attentional and multi-scale feature fusion for augmented representation power. Our back projection method recalibrates the current estimation by establishing feedback connections between high-level and low-level attributes in an attentional and discriminative manner. Second, we propose to decompose the input image and separately process the distinct frequency components, whose derived latents are recombined using a novel dual attention module, so that details inside regions of interest could be explicitly manipulated. Third, we propose a novel training scheme for reducing the latent rounding residual. Experimental results show that, when measured in PSNR, our model reduces BD-rate by 9.88% and 10.32% over the state-of-the-art method, and 4.12% and 4.32% over the latest coding standard Versatile Video Coding (VVC) on the Kodak and CLIC2020 Professional Validation dataset, respectively. Our approach also produces more visually pleasant images when optimized for MS-SSIM. The significant improvement upon existing methods shows the effectiveness of our method in preserving and remedying spatial information for enhanced compression quality.

1. Introduction

Lately, the demand for image compression has increased dramatically to cope with the enormous amount of high-resolution images produced by modern devices. Based on deep neural networks (DNNs), neural image compression has reinvigorated this domain with its superb capacity to learn in a data- and metric-driven manner, as opposed to their conventional, handcrafted counterparts [17].

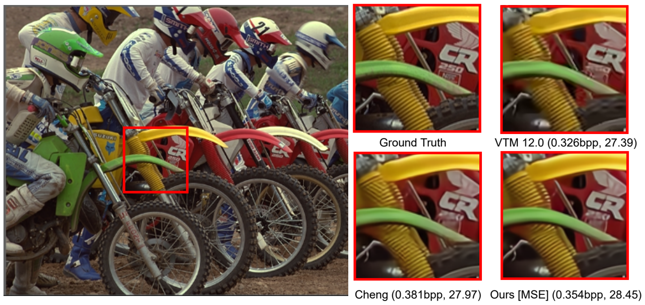


Figure 1. Comparison of *kodim05.png* reconstructed by different methods. The image is cropped for convenient visualization. Notice that the tilted, shadowy artifacts in the yellow tube region have been largely suppressed by our method.

Neural image compression typically employs autoencoders to model the compression and reconstruction process as a unified task and optimize the rate-distortion tradeoff jointly. Such methods map the input image into a more compact latent intermediate via an encoder and inversely transform the quantized latent back to generate the reconstructed image on the decoder side. Many researches concentrate on optimizing the network architecture, e.g., GDN [6], residual blocks [34, 23], RNNs [35, 22, 36], to reduce bit-rates with alleviated quality degradation. Meanwhile, some other works focus on reducing the entropy of the latent representations to attain fewer encoding bits. Earlier works [8, 34] in this respect incorporate elementwise entropy models to encode each element independently. Later advancements introduce hierarchical hyperprior networks [7] and autoregressive components [20, 25] into the VAE framework to explicitly estimate the entropy of the latent representation by utilizing prior information. Currently, the rate-distortion performance of the state-of-the-art methods has surpassed that of reigning compression codecs, such as BPG [9] and VVC [33], in both PSNR and MS-SSIM.

Nonetheless, existing schemes remain limited in faithfully restoring the original image from its compact representation. The reconstructions at low compression rates tend to be over-smoothed and contain undesirable artifacts. One major issue of the autoencoder is that, while it excels at extracting contextualized, non-linear information for effective decorrelation, it stumbles in preserving spatial image details that are crucial to faithful reconstruction, since down-sampling via convolution layers is inherently non-injective due to the loss of high-frequency details. Another limitation of current implementations is that the input image is usually compressed in its RGB format, in which the easily-lost high-frequency details are mingled with large-scale variations. The inability to distinguish distinct frequency characteristics makes it even harder for the network to preserve or infer fine-grained details for optimal reconstruction.

In this paper, to enable mutual facilitation between low- and high-level image properties, we replace the standard feedforward up- and down-sampling layers with a novel Attentional Multi-scale Back Projection (AMBP) module. Our AMBP module efficiently aggregates intermediate features from higher to lower layers of the network, allowing it to attain semantically rich features, on the one hand, and extrapolate fine spatial details, on the other. Retaining the desired properties of both gives the network greater flexibility to decide which information should be preserved for better rate-distortion trade-offs. To extract richer visual representations, we leverage channel attention and a soft attention mechanism that consolidates the input feature maps in a weighted average fashion.

Moreover, we propose to extract and process the distinctive frequency components of the input image via frequency decomposition. In this way, the network could yield further efficiencies in representation by exploiting various pieces of information that carry different frequency characteristics. Our method deploys a dual-branch encoder to compress the separate layer components in parallel and later recombines their derived latents using a novel dual attention module. Besides, to reduce the quantization residual of the latent, we modify the mixed training scheme [26] by adding a rounding loss of the latent, which enforces the network to focus on reducing the quantization error whilst optimizing for the final reconstruction. The main contributions of this work are:

- A novel back projection approach capable of producing contextualized outputs with enriched details via multi-scale context aggregation across stages.
- An effective scheme that decomposes the image into distinct frequency components, processes them separately, and recombines the results via a dual attention module to yield the latent representation.
- A finetuning strategy for reducing the error caused by rounding the latent to facilitate reconstruction.

2. Related Work

2.1. Conventional Image Compression

Conventional compression standards, such as JPEG [37], JPEG2000 [30], BPG [9] and VVC [33], are handcrafted pipelines that rely on manual tuning, which requires extensive expertise and is extremely time-consuming. These schemes transform the input image into compressed coefficients, apply quantization to prune the least informative bits, and entropy encode the quantized coefficients into bitstream files. Moreover, some hybrid techniques [18, 16] have been developed, which apply learned image restoration methods to remove the undesirable artifacts from images reconstructed by conventional codecs. Nonetheless, such hybrid methods still suffer from blocking effects and cannot be jointly optimized via the automated process, which hampers the development of more sophisticated architectures.

2.2. Neural Image Compression

Network Architecture Design. Neural image compression has achieved some major breakthroughs in the past few years. Since the early attempts by Toderici *et al.* [35] to utilize convolutional LSTMs for image compression, considerable improvements have been made in incorporating tailored modules for neural image compression. Ballé *et al.* [6] put forward a nonlinear normalization technique called generalized divisive normalization (GDN), which demonstrates impressive capacities in decorrelating data from natural images. Zhang *et al.* [42] exploit the expressive power of residual connections and propose a non-local attention block to capture the global dependencies between latent elements. Some works [39, 22] resort to recurrent structures to remove the spatial redundancy between parts of the neural images, where each previous part serves as a reference for the current part. Recent efforts [4, 31, 32, 24, 27, 29, 28] in abridging the huge gap between human perceptual preferences and the dominating distortion metrics have also achieved remarkable progress in helping the networks reconstruct images that are more perceptually convincing.

Quantization. The extracted latent is usually discretized via quantization to support lossless entropy coding. Many studies adopt additive uniform noises [8] to simulate the effects of quantization within a differentiable process, while others either adopt straight-through gradients that propagate the gradient of the identity function or develop a soft quantization technique [2, 23], e.g., learnable clustering

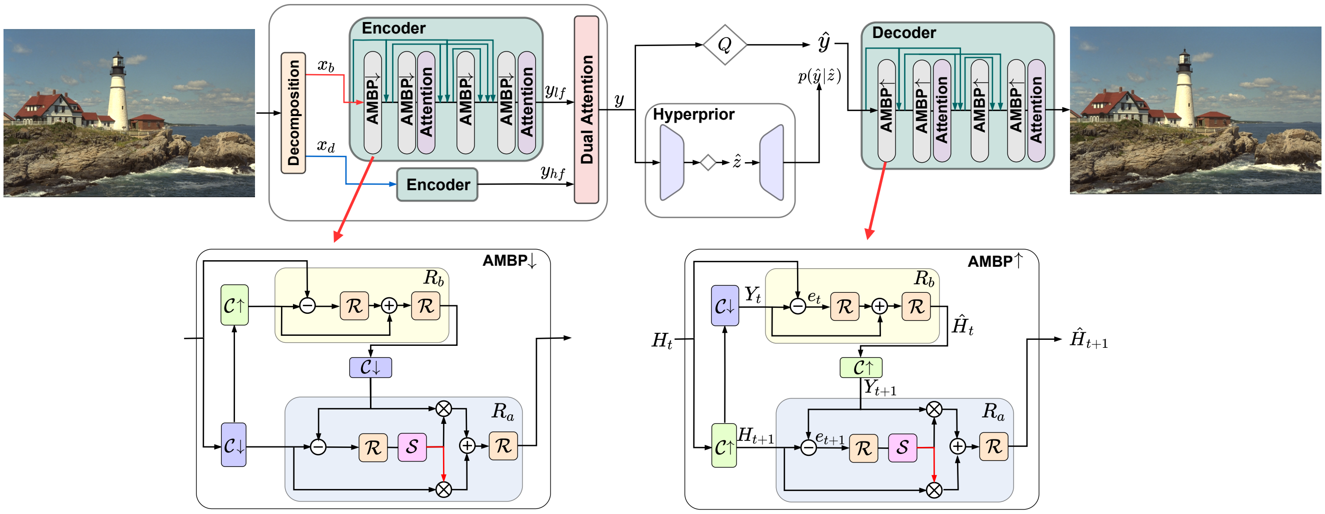


Figure 2. Network architecture of the proposed method. Q denotes the quantization operation. $\hat{y}$ and $\hat{z}$ denote the quantized latent and the quantized side information, respectively. The black and red arrows in the attentional fusion block R_a denote reweighting by W and $(1-W)$, respectively, where W denotes the attention weights learned by channel attention module $\mathcal{S}$. $\otimes$ denotes element-wise multiplication. $\mathcal{C}\uparrow$ and $\mathcal{C}\downarrow$ denote down-sampling and up-sampling, implemented by strided convolutions and subpixel convolutions (activation = LeakyReLU, kernel size = 3x3), respectively. $\mathcal{R}$ denotes the residual block that consists of two 3x3, ReLU-activated convolution layers with a skip connection.

and nearest neighbor assignment, to reduce the round-off residual. Meanwhile, how to correctly predict the quantization residual is an actively studied topic. For instance, Dumas *et al.* [12] propose a model that optimizes the quantization step size for each feature map of the latent, and Minnen *et al.* [26] condition the latent rounding residual on the hyperprior and the already-decoded latent slices for more accurate prediction.

Entropy Model. Entropy coding occupies more bits to encode elements that have a smaller probability of occurrences. Many works concentrate on obtaining a more accurate entropy estimation of the latent representations. The pioneering works by Toderci *et al.* [35] and Ballé *et al.* [8] develop a fully factorized entropy model to predict the probability distribution of the latent and independently encode each element with arithmetic coding. Expanding on this design, Ballé *et al.* [7] introduce hyperprior to competently learn the pixel-wise dependencies of the latent, where the distribution is approximated by an isotropic Gaussian with standard deviation σ. Similar to Lee *et al.* [20], Minnen *et al.* [25] improve the hyperprior by estimating both mean and standard deviation of the learned latent's distribution and incorporate an autoregressive context model that explicitly conditions each element on previously decoded elements to further reduce the spatial redundancy between adjacent pixels. Later studies [10, 21] augment the context model by utilizing more complex distributions and incorporating other types of correlations.

However, most of the existing neural image compression methods fail to make efforts to retain both low- and high-level features as the computation forwards or consider the frequency entanglement issue, which will be discussed further in **Section 3.3**.

3. Method

3.1. Overall Framework

Problem Formulation. The objective of neural image compression is to achieve minimal distortion of the restored image under a specific rate constraint. Given an input image x, the encoder E squeezes out the spatial redundancies in it and generates the latent intermediate y. The latent y is then quantized using the quantization function Q to attain the discrete code $\hat{y}$, from which the reconstructed image $\hat{x}$ is generated. The complete process can be formulated as:

$$\begin{aligned} y &= E(x;\phi) \\ \hat{y} &= Q(y) \\ \hat{x} &= D(\hat{y};\theta), \end{aligned} \tag{1}$$

where ϕ and θ denote the trainable parameters of the encoder E and decoder D, respectively. The rate term R represents the required number of bits to encode $\hat{y}$, and to more accurately estimate the entropy of latent code $\hat{y}$, we parameterize its true distribution $p_{\hat{y}}$ using an entropy model $P(\hat{y})$ with Gaussian Mixture Likelihoods and an autoregressive context model. Here, R can be formulated as the cross-entropy of $p_{\hat{y}}$ and $P(\hat{y})$, which is minimized when two distributions match:

$$R = \mathbb{E}_{\hat{y}\sim p_{\hat{y}}}[-\log P(\hat{y})]. \tag{2}$$

Compression and quantization incur a distortion $d(x, \hat{x})$ that is usually measured by PSNR or MS-SSIM. Formulating E, D and $P(\hat{y})$ as neural networks allows them to be optimized jointly by minimizing the rate-distortion trade-off $\mathcal{L}$:

$$\mathcal{L} = \lambda \cdot d(x, \hat{x}) + R, \tag{3}$$

where λ controls the trade-off.

Network Architecture. As shown in Fig.2, the encoder side of our design consists of a decomposition module, a dual-branch encoder, and a dual attention module. Instead of processing the input image in its RGB format, we propose to extract its low- and high-frequency layer components and compress them separately using the dual-branch encoder. The two identical branches consist of four AMBP↓ modules responsible for down-sampling with two spatial attention blocks [10] in between. We adopt the dense connectivity from [13], meaning that the current AMBP module process the concatenated outputs of all previous modules. The down-sampled latents of the frequency layers are then rescaled and combined to form the complete latent representation y via the dual attention module. The hyperprior model and the context model of our network follow the same design as in [10]. The single-branch decoder is the mirror reflection of the encoder's branch, consisting of four AMBP↑ modules for up-sampling and two spatial attention blocks in between. The decoder up-samples the quantized latent $\hat{y}$ to yield the reconstruction image $\hat{x}$.

3.2. Attentional Multi-scale Back Projection

Back projection was first put forward in DBPN [13] for image super-resolution. The back projection technique iteratively utilizes the feedback residual to refine high-resolution (HR) images, based on the assumption that the projected, down-sampled version of a super-resolution image should be as close to the original low-resolution (LR) image as possible. We adopt and extend this technique to solve image compression problems and construct our building blocks entitled AMBP. Specifically, we replace the standard convolution and deconvolution (or subpixel convolution) layers with AMBP↓ and AMBP↑, respectively.

Convolution layers of the autoencoder trade fine spatial details for copious semantic information after repetition of down-sampling operations, making it less reliable for faithful image reconstruction. To address this issue, AMBP aggregates multi-scale features across stages in a trainable way. That is, the current stage features are consolidated by the complementary information (spatially accurate or contextually rich) from later computations. The refined feature maps in turn produce features of higher quality in the next stage, thereby achieving progressive improvement to the intermediate features that propagate throughout the computation. Diversifying the contexts also empowers the network with greater flexibility in selecting the important portion of information to be retained.

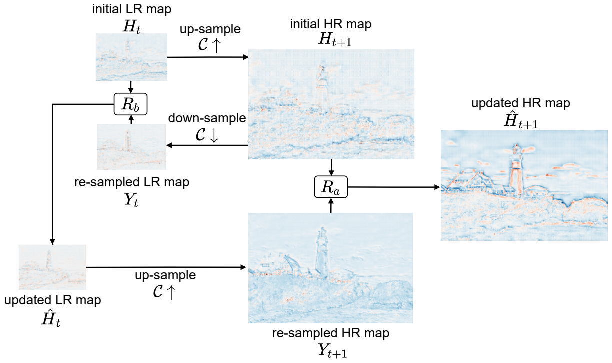


Figure 3. Illustration of the back projection procedure using feature maps sampled from the decoder when reconstructing *kodim21.png*. The updated HR map $\hat{H}_{t+1}$ contains better defined details than the initial HR map H_{t+1}. Here, $\mathcal{C}\uparrow$ and $\mathcal{C}\downarrow$ denote down-sampling and up-sampling, resepctively.

Taking up-sampling as an example (Fig. 3), our AMBP↑ module refines an HR map H_{t+1}, up-sampled from H_t, by applying reverse mapping to recover its original resolution. H_t contains multi-scale information from previous layers due to the dense connections. Despite having the same resolution, the re-sampled feature map Y_t encloses details that were not priviously available to H_t. These details are then integrated into H_t using a fusion module R_b, producing an updated LR map $\hat{H}_t$ which is up-sampled again by $\mathcal{C}\uparrow$ to yield a re-sampled HR map Y_{t+1}. To facilitate in-scale feature fusion, we leverage an attentional fusion module R_a that aggregates H_{t+1} and Y_{t+1} and update the former as $\hat{H}_{t+1}$ that contains finer details. The described process can be written as:

$$\begin{aligned}
Y_t &= \mathcal{C}\downarrow(H_{t+1}) = \mathcal{C}\downarrow(\mathcal{C}\uparrow(H_t)) \\
\hat{H}_t &= R_b(H_t, Y_t) \\
Y_{t+1} &= \mathcal{C}\uparrow(\hat{H}_t) \\
\hat{H}_{t+1} &= R_a(H_{t+1}, Y_{t+1}).
\end{aligned} \tag{4}$$

Specifically, the feature fusion is based on residual calculation, rather than addition or concatenation. As shown in Fig. 2, the residual fusion module R_b adaptively aggregates H_t and Y_t according to their residual $e_t = H_t - Y_t$. Intuitively, e_t represents distinctive information available in one source while missing in the other. We further incorporate the attentional fusion module R_a for the (H_{t+1}, Y_{t+1}) pair. Instead of processing the residual with residual blocks $\mathcal{R}$ as in R_b, we complement the spatial attention blocks with channel attention [15] to enhance the modelling capacity of R_a. Similar to [11], we adaptively aggregate the information carried by e_{t+1} using a soft fusion scheme that reweights the respective inputs by W

and $(1-W)$, where W is the normalized attention map. In this way, the network subtly performs importance weighing between the two inputs without explicitly learning two sets of weights. The back projection procedure is formulated as:

$$\begin{gathered}R_b(H_t, Y_t) = \mathcal{R}(Y_t + \mathcal{R}(e_t)) \\ W = \mathcal{S}(\mathcal{R}(e_{t+1})) \qquad (5) \\ R_a(H_{t+1}, Y_{t+1}) = \mathcal{R}(W \otimes H_{t+1} + (1-W) \otimes Y_{t+1}),\end{gathered}$$

where $\mathcal{R}$ denotes residual blocks, W denotes the attention map, $\mathcal{S}$ denotes channel attention, and $\otimes$ denotes element-wise multiplication.

The benefits of our proposed AMBP module are three-fold. First, it further optimizes the original feature map H_{t+1} and facilitates both in-scale and cross-scale feature fusion without necessarily relying on iterations. Second, the proposed soft content selection scheme enables more adaptive feature fusion by implicitly balancing the weighing of the source inputs before aggregation. Third, the feature fusion based on residuals allows the network to focus only on distinctive information, making the gradient update better guided and more efficient, so incorporating another residual-based fusion operation could further stabilize and accelerate the training procedure.

3.3. Frequency Decomposition

Frequency Decomposition Module. Most natural images contain prolific frequency attributes that are, however, intertwined and therefore hard to extract. Hence, we believe that greater adaptivity could be attained by decomposing an image into several layer components of different frequency attributes, i.e., the base layer and the detail layer [18, 40]. With the decomposed signals, improved flexibility could be achieved from manipulating the layer components separately and recombining them to yield the final result. Further, the high-frequency components lost during down-sampling, as pointed out by Nyquist-Shannon sampling theorem, could now be explicitly manipulated by the network so that details inside intended regions could be better retained during the detail layer pass.

As illustrated in Fig.4, the low-frequency components across scales are obtained using average pooling with various kernel sizes. The high-frequency components are attained by subtracting the corresponding low-frequency component from the input image x. To produce the base layer x_b, we pass the concatenated low-frequency components to a residual block $\mathcal{R}$. The detail layer x_d containing high-frequency information is attained in a similar manner. As the original image x also contains rich information, it is concatenated with x_b and x_d, which are then processed separately by the dual-branch encoder. The dual-branch encoder progressively down-samples the respective layer components into their latent representations y_{lf} and y_{hf}.

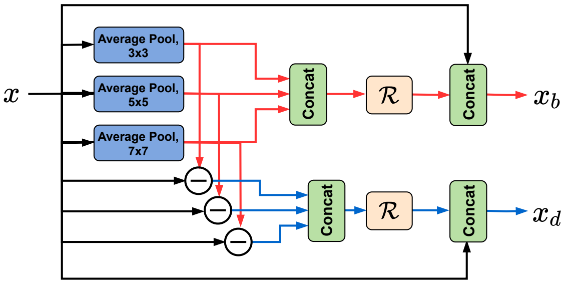


Figure 4. The frequency decomposition module, where the red and blue arrows denote the low- and high-frequency components, respectively. $\mathcal{R}$ denotes the residual block.

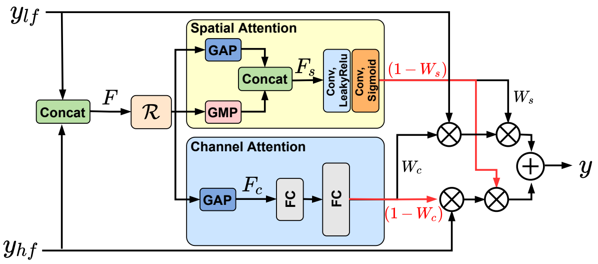


Figure 5. Dual attention module. FC denotes fully connected layers. The black and red arrows indicate multiplying the feature map by the corresponding attention weight W and (1-W), respectively. GAP and GWP denote global average pooling and global max pooling, respectively. $\mathcal{R}$ denotes the residual block.

Dual Attention. The latents y_{lf} and y_{hf} are aggregated then using a dual attention module (as shown in Fig. 5), which is adopted to facilitate information sharing along both dimensions. The latents of the respective frequency layers are concatenated along the channel dimension to produce feature map F, which is then transformed by a residual block and passed to the channel and the spatial attention module. To reduce computation, the spatial attention module independently applies global average pooling and global max pooling to F along the channel dimension and concatenates the results to form feature map $F_s \in \mathbb{R}^{H\times W\times 2}$, from which is the spatial attention map $W_s \in \mathbb{R}^{H\times W\times 1}$ extracted. The channel attention feature map $W_c \in \mathbb{R}^{1\times 1\times C}$ is generated using SE blocks [15]. We adopt the soft selection trick to improve representations as well. The low-frequency latent y_{lf} is rescaled by W_c and then W_s while the high-frequency latent y_{hf} is rescaled by $(1-W_c)$ and then $(1-W_s)$. The re-weighted latents are then summed to yield the final latent representation y.

3.4. Mixed Training Scheme

Following the work [26], we adopt noisy relaxation to approximate quantization to jointly optimize the net-

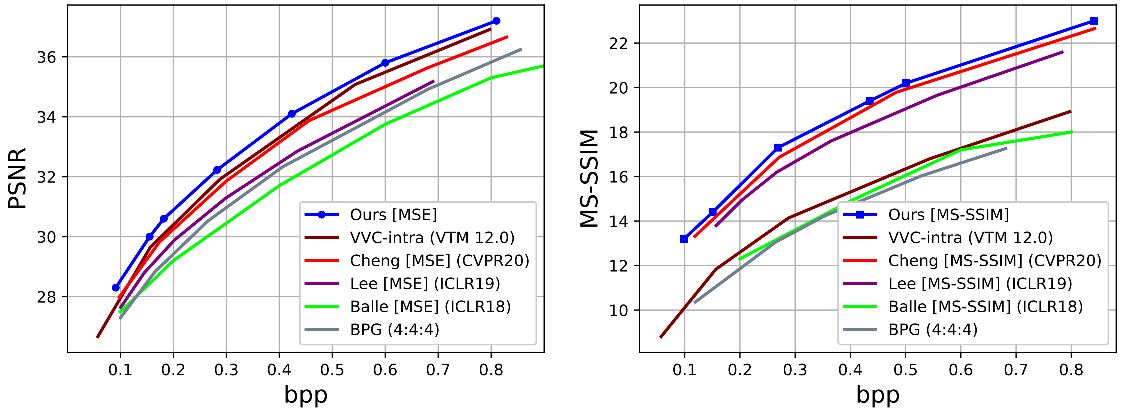


Figure 6. Performance evaluation on the Kodak dataset. Our method yields improved coding performance than existing learning-based methods and VVC-intra [33].

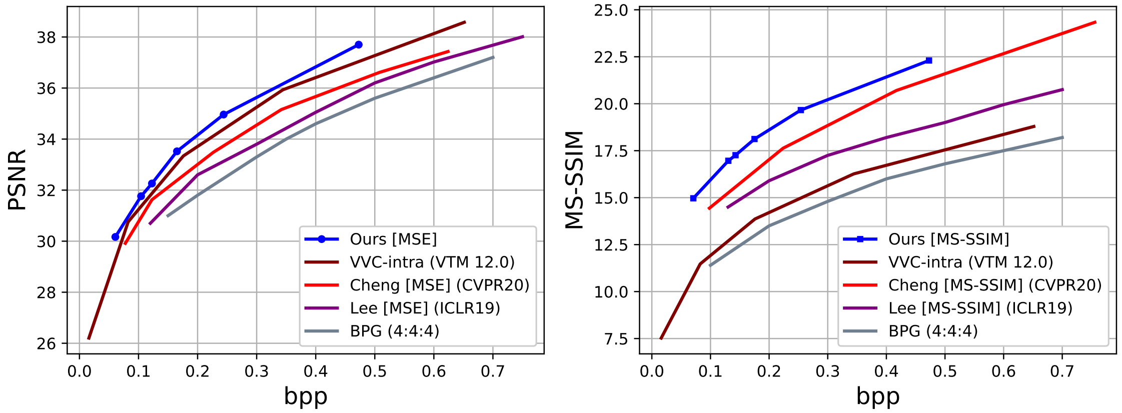


Figure 7. Comparison of rate-distortion performance on the CLIC Professional Validation dataset [1]. Our method outperforms existing learning-based methods and VVC-intra [33].

work and then finetune the decoder D with rounding representation and an additional rounding loss term $d_r = MSE(y, \tilde{y})$, where $\tilde{y}$ denotes the latent refined by the first AMBP↑ module on the decoder side. We notice that the reconstruction quality could be considerably enhanced by decoding the original rather than quantized latent, even without further tuning, so we incorporate this constraint into the loss function, which enforces the network to focus on reducing the latent rounding residual while optimizing for the quality of the final reconstructed image. The loss function $\mathcal{L}_f$ for finetuning is:

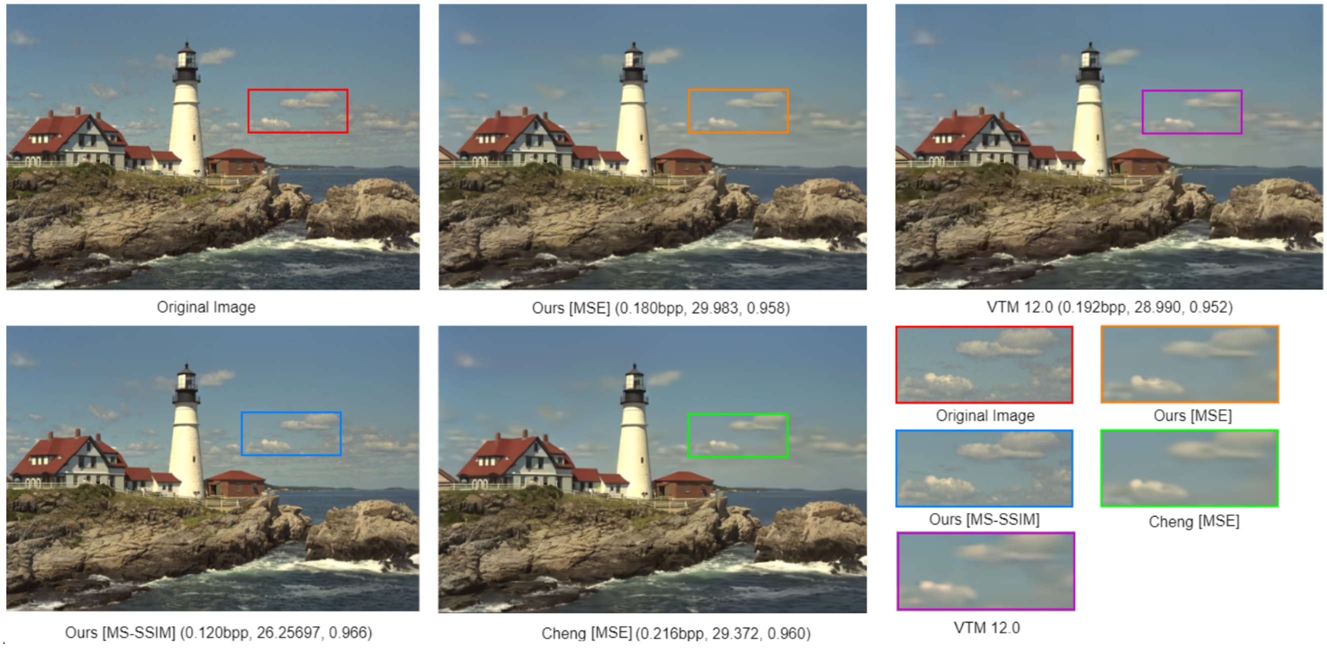
$$\mathcal{L}_f = d(x, \hat{x}) + \beta \cdot d_r, \tag{6}$$

where β controls the weight of the rounding loss term.

4. Experimental Results

4.1. Implementation and Training Details

We trained the proposed networks using cropped images of size 256x256 from DIV2K [3] , Flickr2K[3], and CLIC training dataset [1] without augmentation. The weights of two identical branches of the encoder are shared to reduce the model complexity. We used the Adam algorithm to jointly optimize the networks for *1.5M* steps with a mini-batch size of 4. The initial learning rate was set to 1×10^{-4} and halved every *5k* steps for the last *300k* steps. After that, we finetuned the sub-modules responsible for reconstruction (i.e., the decoder) for the objective described by Eq.(6) for *500k* steps, where the initial learning rate was set to 5×10^{-5} and halved every *100k* steps. The networks were optimized for MSE and MS-SSIM, respectively. When optimized for MSE, the value of λ belongs to the set $\{0.0015, 0.0032, 0.004, 0.0075, 0.015, 0.03, 0.05\}$ and the channel number was set to 128 for the four lower-rate networks and 192 for the three higher-rate networks. When optimized for MS-SSIM, the value of λ belongs to the set $\{3, 4.5, 12, 32, 45, 120\}$, where the channel number was set to 128 for the four lower-rate and 192 for the two higher-rate networks. The distortion d is defined as d=1-MS-SSIM($\boldsymbol{x}$,$\hat{\boldsymbol{x}}$). The coefficient β was set to 1 and 0.01 for MSE- and MS-SSIM-optimized models, respectively.


Original Image

Ours [MSE] (0.180bpp, 29.983, 0.958)

VTM 12.0 (0.192bpp, 28.990, 0.952)

Ours [MS-SSIM] (0.120bpp, 26.25697, 0.966)

Cheng [MSE] (0.216bpp, 29.372, 0.960)

Original Image

Ours [MSE]

Ours [MS-SSIM]

Cheng [MSE]

VTM 12.0

Figure 8. Comparison of *kodim21.png* reconstructed by different methods. The details of the cloud are well preserved using our model optimized for MS-SSIM, and the image reconstructed using our MSE-optimized attains comparable visual quality with VVC-intra and the reference network [10].

4.2. Performance Comparison

We first evaluated our networks by obtaining the average rate-distortion performance in terms of PSNR and MS-SSIM on the commonly used Kodak PhotoCD dataset [19], which contains 24 high-quality images. The rate-distortion (RD) curves are plotted in Fig.6, where the rate is measured by bits per pixel (bpp). Note that MS-SSIM is converted to decibels, in accordance with the previous works [10, 21], so that differences in performance are more distinguishable. It can be seen from the results that our model consistently outperforms both the state-of-the-art deep learning-based methods and the advanced compression standard VVC-intra (VTM 12.0) for all rates measured. We further measured the reductions in BD-rate of our model, which is defined as the average saving in bitrate between two models for a given quality metric. Regarding PSNR, the average reduction in BD-rate of our method on the Kodak dataset are 9.88% and 4.12% against the current state-of-the-art model [10] and the VVC-intra, respectively.

Moreover, we assessed the effectiveness of our method on two high-resolution datasets, the CLIC Professional Validation set [1] and the Tecnick dataset [5]. As illustrated in Fig. 7, our method yields better coding performance than previous methods and VVC-intra in terms of both PSNR and MS-SSIM. Regarding PSNR, the average reduction in BD-rate of our model against the current state-of-the-art model [10] and the VVC-intra is 10.32% and 4.32%, respectively, on the CLIC Professional validation dataset. Please refer to the supplementary materials for comparisons of RD curves regarding PSNR on the Tecnick dataset.

Our proposed method also attains desirable visual quality. Fig.1 and Fig.8 show the reconstructed images *kodim05.png* and *kodim21.png* by various compression methods. As shown in the enlarged part of Fig.1, the color and edges are better restored by our method, and the tilted, shadow-like artifacts in the yellow tube region in the reconstructed images of other methods are largely suppressed by that of ours. Further, as shown in Fig.8, the textures of the cloud are well preserved using our MS-SSIM-optimized model, and the reconstruction by our MSE-optimized model yields comparable visual quality with VVC-intra and the reference image compression network [10]. We also quantitatively evaluated our MSE-optimized model regarding LPIPS [41] and verified that our method attains better LPIPS scores than VVC-intra on all three datasets tested. For the RD curve plots regarding LPIPS and more visual comparisons of reconstructed images, please refer to the supplementary materials.

4.3. Complexity Analysis

As shown in Table 1, our model has 2.27 times more parameters than that of the reference model [10]. On average, our encoding time and decoding time are about 2.57 times and 1.64 times longer, respectively, with the same hardware configuration. The larger increase in latency on the encoder side attributes to the fact that, in practice, the branches of the encoder are executed sequentially instead of in parallel.

Table 1. Number of parameters, average encoding and decoding time of our model against the reference model [10] on the Kodak dataset for low-bit image compression.

	No. of Params	Encoding (s)	Decoding (s)
Ours	25.4M	104.24	43.12
Ref [10]	11.2M	40.52	26.28

4.4. Ablation Study

We present ablative experiments to analyze the contribution of each component of our model. We ablated the design choices and measured the average increase in BD-rate on the Kodak dataset. The followings can be summarized from the ablation results:

AMBP. As shown in Table 2, the model suffers from the greatest performance drop after discarding AMBPs. Replacing the soft selection with a single set of channel attention weights yields a 1.21% increase in BD-rate. The complexity analysis shows that our model has considerably more parameters than the reference model [10], so we also replaced AMBPs with the 3-iteration DBPN modules [13] to make the number of parameters comparable. We observed a 1.17% and 1.98% increase in BD-rate from ablating AMBP↓ and AMBP↑, respectively, which further validates that the architectural modifications we made to the back projection methods are effective.

Table 2. Ablative analysis of AMBPs by measuring the average increase in BD-rate.

AMBP↓	AMBP↑	Soft Selection	**BD-rate↑**
✗	✗	✗	**8.61%**
✔	✔	✗	1.21%
DBPN [13]	✔	✔	1.17%
✔	DBPN [13]	✔	1.98%

Frequency Decomposition. Table 3 indicates that replacing base layer x_b and detail layer x_d with the original image increases the BD-rate by 2.08%, which validates that decomposing and separately processing the distinctive frequency components of the input image are beneficial to improving the coding efficiency. We further ablated the specific design choices, including concatenating the original image and adding the residual block, for the frequency decomposition module and verify their effectiveness according to the increased BD-rate. To ablate the dual attention module, we replaced it with stacks of four convolution layers and attained a 2.37% increase in BD-rate.

Rounding Loss. We show that removing the rounding loss for mixed training from the full model leads to deteriorated coding efficiency. Excluding the proposed rounding loss from the finetuning process increases the BD-rate by 3.52%. We visualize the effect of incorporating the rounding loss by sampling the top four feature maps with the greatest discrepancy after rounding from the original latent and plotting the pixel-wise absolute error. As shown in Fig. 9, the refined feature maps with rounding loss added are much less deviated than those without.

Table 3. Ablative analysis of design choices for frequency decomposition. Here, Concat refers to concatenating the original image to the low- & high-frequency components, and ResBlock denotes the residual block $\mathcal{R}$ in the decomposition module.

Base/Detail	Concat	ResBlock	Dual Attention	**BD-rate↑**
Original Image	✔	✔	✔	2.08%
✔	✗	✔	✔	0.09%
✔	✔	✗	✔	0.15%
✔	✔	✔	✗	**2.37%**

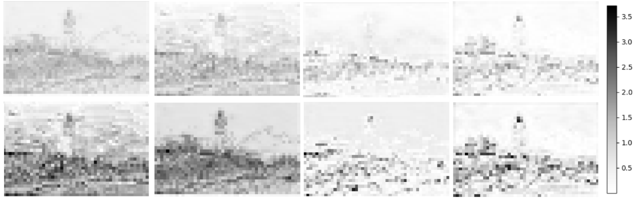


Figure 9. Visualization of latent residual maps with (*upper row*) and without (*lower row*) rounding loss during finetuning ($\lambda = 0.004$). Imposing the latent rounding loss effectively reduces the residual without modifying the network architecture.

5. Conclusion

In this paper, we propose a neural image compression scheme using a novel AMBP module and frequency decomposition. We reformulate the iterative projection operations into a multi-scale feature fusion module and incorporate channel attention with soft content selection. We also propose a novel frequency decomposition method that enables the network to focus on distinct frequency components of the input image, where their derived latents are adaptively rescaled and integrated using an efficient dual attention module. Further, we adopt a novel training scheme that exploits upsampled results to reduce the residual caused by rounding the latent. Experimental results show that our method outperforms the existing neural compression frameworks and the next-generation compression standard VVC-intra by a noticeable margin.

Acknowledgments

Yuchao Dai was supported in part by National Natural Science Foundation of China (61871325) and National Key Research and Development Program of China (2018AAA0102803). We would like to thank the anonymous reviewers and the area chairs for their useful feedback.

References

[1] Workshop and challenge on learned image compression. https://www.compression.cc/, 2020. 6, 7

[2] Eirikur Agustsson, Fabian Mentzer, Michael Tschannen, Lukas Cavigelli, Radu Timofte, Luca Benini, and Luc Van Gool. Soft-to-hard vector quantization for end-to-end learning compressible representations. In *Advances in Neural Information Processing Systems*, pages 1141–1151, 2017. 2

[3] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR) Workshops*, pages 126–135, 2017. 6

[4] Eirikur Agustsson, Michael Tschannen, Fabian Mentzer, Radu Timofte, and Luc Van Gool. Generative adversarial networks for extreme learned image compression. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 221–231, 2019. 2

[5] Nicola Asuni and Andrea Giachetti. Testimages: a large-scale archive for testing visual devices and basic image processing algorithms. In *STAG*, pages 63–70, 2014. 7

[6] Johannes Ballé, Valero Laparra, and Eero P Simoncelli. Density modeling of images using a generalized normalization transformation. In *International Conference on Learning Representations*, 2016. 1, 2

[7] Johannes Ballé, David Minnen, Saurabh Singh, Sung Jin Hwang, and Nick Johnston. Variational image compression with a scale hyperprior. In *International Conference on Learning Representations*, 2018. 1, 3

[8] Johannes Ballé, Valero Laparra, and Eero Simoncelli. End-to-end optimized image compression. In *International Conference on Learning Representations*, 2017. 1, 2, 3

[9] Fabrice Bellard. Bpg image format. https://bellard.org/bpg. 2014. 2

[10] Zhengxue Cheng, Heming Sun, Masaru Takeuchi, and Jiro Katto. Learned image compression with discretized gaussian mixture likelihoods and attention modules. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7939–7948, 2020. 3, 4, 7, 8

[11] Yimian Dai, Fabian Gieseke, Stefan Oehmcke, Yiquan Wu, and Kobus Barnard. Attentional feature fusion. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3560–3569, 2021. 4

[12] Thierry Dumas, Aline Roumy, and Christine Guillemot. Autoencoder based image compression: can the learning be quantization independent? In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1188–1192. IEEE, 2018. 3

[13] Muhammad Haris, Gregory Shakhnarovich, and Norimichi Ukita. Deep back-projection networks for super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1664–1673, 2018. 4, 8

[14] Leonhard Helminger, Abdelaziz Djelouah, Markus Gross, and Christopher Schroers. Lossy image compression with normalizing flows. In *arXiv preprint arXiv:2008.10486*, 2020.

[15] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7132–7141, 2018. 4, 5

[16] Yueyu Hu, Haichuan Ma, Dong Liu, and Jiaying Liu. Compression artifact removal with ensemble learning of neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2020. 2

[17] Yueyu Hu, Wenhan Yang, Zhan Ma, and Jiaying Liu. Learning end-to-end lossy image compression: A benchmark. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 1

[18] Soo Ye Kim, Jihyong Oh, and Munchurl Kim. Deep sr-itm: Joint learning of super-resolution and inverse tone-mapping for 4k uhd hdr applications. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3116–3125, 2019. 2, 5

[19] Eastman Kodak. Kodak lossless true color image suite. http://r0k.us/graphics/kodak. 1993. 7

[20] Jooyoung Lee, Seunghyun Cho, and Seung-Kwon Beack. Context-adaptive entropy model for end-to-end optimized image compression. In *International Conference on Learning Representations*, 2018. 1, 3

[21] Jooyoung Lee, Seunghyun Cho, and Munchurl Kim. An end-to-end joint learning scheme of image compression and quality enhancement with improved entropy minimization. In *arXiv preprint arXiv:1912.12817*, 2019. 3, 7

[22] Chaoyi Lin, Jiabao Yao, Fangdong Chen, and Li Wang. A spatial rnn codec for end-to-end image compression. In *International Conference on Computer Vision and Pattern Recognition*, 2020. 1, 2

[23] Fabian Mentzer, Eirikur Agustsson, Michael Tschannen, Radu Timofte, and Luc Van Gool. Conditional probability models for deep image compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4394–4402, 2018. 1, 2

[24] Fabian Mentzer, George Toderici, Michael Tschannen, and Eirikur Agustsson. High-fidelity generative image compression. In *Advances in Neural Information Processing Systems*, 2020. 2

[25] David Minnen, Johannes Ballé, and George Toderici. Joint autoregressive and hierarchical priors for learned image compression. In *Advances in Neural Information Processing Systems*, pages 10794–10803, 2018. 1, 3

[26] David Minnen and Saurabh Singh. Channel-wise autoregressive entropy models for learned image compression. In *IEEE International Conference on Image Processing*, pages 3339–3343. IEEE, 2020. 2, 3, 5

[27] Yash Patel, Srikar Appalaraju, and R Manmatha. Deep perceptual compression. *arXiv preprint arXiv:1907.08310*, 2019. 2

[28] Yash Patel, Srikar Appalaraju, and R Manmatha. Human perceptual evaluations for image compression. *arXiv preprint arXiv:1908.04187*, 2019. 2

[29] Yash Patel, Srikar Appalaraju, and R Manmatha. Saliency driven perceptual image compression. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 227–236, 2021. 2

[30] Majid Rabbani and Rajan Joshi. An overview of the jpeg 2000 still image compression standard. *Signal processing: Image communication*, 17(1):3–48, 2002. 2

[31] Oren Rippel and Lubomir Bourdev. Real-time adaptive image compression. In *International Conference on Machine Learning*, pages 2922–2930. PMLR, 2017. 2

[32] Shibani Santurkar, David Budden, and Nir Shavit. Generative compression. In *Picture Coding Symposium*, pages 258–262. IEEE, 2018. 2

[33] Gary Sullivan and Jens-Rainer Ohm. Versatile video coding. *JVET-T2002*, 2020. 2, 6

[34] Lucas Theis, Wenzhe Shi, Andrew Cunningham, and Ferenc Huszár. Lossy image compression with compressive autoencoders. In *International Conference on Learning Representations*, 2017. 1

[35] George Toderici, Sean M. O'Malley, Sung Jin Hwang, Damien Vincent, David Minnen, Shumeet Baluja, Michele Covell, and Rahul Sukthankar. Variable rate image compression with recurrent neural networks. In *International Conference on Learning Representations*, 2016. 1, 2, 3

[36] George Toderici, Damien Vincent, Nick Johnston, Sung Jin Hwang, David Minnen, Joel Shor, and Michele Covell. Full resolution image compression with recurrent neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5306–5314, 2017. 1

[37] Gregory K Wallace. The jpeg still picture compression standard. *IEEE transactions on consumer electronics*, 38(1):xviii–xxxiv, 1992. 2

[38] Yaolong Wang, Mingqing Xiao, Chang Liu, Shuxin Zheng, and Tie-Yan Liu. Modeling lost information in lossy image compression. In *arXiv preprint arXiv:2006.11999*, 2020.

[39] Maurice Weber, Cedric Renggli, Helmut Grabner, and Ce Zhang. Observer dependent lossy image compression. In *German Conference on Pattern Recognition*, 2020. 2

[40] Ke Xu, Xin Yang, Baocai Yin, and Rynson WH Lau. Learning to restore low-light images via decomposition-and-enhancement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2281–2290, 2020. 5

[41] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 586–595, 2018. 7

[42] Yulun Zhang, Kunpeng Li, Kai Li, Bineng Zhong, and Yun Fu. Residual non-local attention networks for image restoration. In *International Conference on Learning Representations*, 2019. 2