MLIC: Multi-Reference Entropy Model for Learned Image Compression

Wei Jiang
jiangwei@stu.pku.edu.cn
Shenzhen Graduate School, Peking University
Shenzhen, China

Jiayu Yang
jiayuyang@pku.edu.cn
Shenzhen Graduate School, Peking University
Peng Cheng Laboratory
Shenzhen, China

Yongqi Zhai
zhaiyongqi@stu.pku.edu.cn
Shenzhen Graduate School, Peking University
Peng Cheng Laboratory
Shenzhen, China

Peirong Ning
peirongning@stu.pku.edu.cn
Shenzhen Graduate School, Peking University
Shenzhen, China

Feng Gao
gaof@pku.edu.cn
School of Arts, Peking University
Beijing, China

Ronggang Wang*
rgwang@pkusz.edu.cn
Shenzhen Graduate School, Peking University
Peng Cheng Laboratory
Migu Culture Technology Co., Ltd
Shenzhen, China

ABSTRACT

Recently, learned image compression has achieved remarkable performance. The entropy model, which estimates the distribution of the latent representation, plays a crucial role in boosting rate-distortion performance. However, most entropy models only capture correlations in one dimension, while the latent representation contains channel-wise, local spatial, and global spatial correlations. To tackle this issue, we propose the Multi-Reference Entropy Model (MEM) and the advanced version, MEM$^+$. These models capture the different types of correlations present in latent representation. Specifically, We first divide the latent representation into slices. When decoding the current slice, we use previously decoded slices as context and employ the attention map of the previously decoded slice to predict global correlations in the current slice. To capture local contexts, we introduce two enhanced checkerboard context capturing techniques that avoids performance degradation. Based on MEM and MEM$^+$, we propose image compression models MLIC and MLIC$^+$. Extensive experimental evaluations demonstrate that our MLIC and MLIC+ models achieve state-of-the-art performance, reducing BD-rate by 8.05% and 11.39% on the Kodak dataset compared to VTM-17.0 when measured in PSNR.

CCS CONCEPTS

• **Computing methodologies → Image compression**.

*Corresponding author.

MM '23, October 29-November 3, 2023, Ottawa, ON, Canada

ACM ISBN 979-8-4007-0108-5/23/10…$15.00
https://doi.org/10.1145/3581783.3611694

KEYWORDS

entropy model; image compression

ACM Reference Format:
Wei Jiang, Jiayu Yang, Yongqi Zhai, Peirong Ning, Feng Gao, and Ronggang Wang. 2023. MLIC: Multi-Reference Entropy Model for Learned Image Compression. In *Proceedings of the 31st ACM International Conference on Multimedia (MM '23), October 29-November 3, 2023, Ottawa, ON, Canada.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3581783.3611694

1 INTRODUCTION

Due to the rise of social media, tens of millions of images are generated and transmitted on the web every second. Service providers need to find more efficient and effective image compression methods to save bandwidth. Although traditional coding methods like JPEG [41], JPEG2000 [9], BPG [7], and VVC [8] have achieved good performance, they rely on manual design for each module which is independent of each other, making joint optimization impossible.

Recently, various learned image compression models [4, 16, 20, 21, 26–30, 36, 37, 40, 44, 46, 47, 51, 58] have been proposed, achieved remarkable performance. Some learned image compression models [11, 12, 34, 39, 56, 57] are already comparable to the advanced traditional method VVC. Most of these models are based on variational autoencoders [24], following the transform, quantization, entropy coding, and inverse transform process. Entropy coding plays an important role in boosting model performance. An entropy model is used to estimate the entropy of latent representation. As it involves estimating the entropy of the latent representation using an entropy model. A powerful and accurate entropy model usually leads to fewer bits.

State-of-the-art learned image compression models [10, 14, 17] usually equip the entropy model with a hyperprior model [5] or a context model [38] to estimate conditional entropy and use conditional probabilities for coding. Context models usually model probabilities and correlations in different dimensions, including local spatial context model, global spatial context model, and channel-wise context model. However, most methods capture conditional probabilities in a single dimension, leading to inaccurate conditional probabilities. To overcome this limitation, we introduce Multi-Reference Entropy Models (MEM) and the enhanced version, MEM+, which effectively capture local spatial, global spatial, and channel-wise contexts. Based on MEM and MEM+, we propose MLIC and MLIC+, which achieve state-of-the-art performance. In our approach, we divide the latent representations into several slices. When compressing a slice, the previously compressed slices are treated as its contexts. A channel-wise context module is adopted to get the channel information from these channel-wise contexts. We conduct local and global context modeling for every slice separately. An auto-regressive local context model leads to serial decoding, while a checkerboard context model [18] can achieve two-pass parallel decoding which divides latent representations into the anchor and non-anchor parts. However, the checkerboard context model can result in up to 4% performance degradation. To address this issue, we propose two different methods: stacked checkerboard context modeling and overlapped checkerboard window attention. Some previous methods focused on global context modeling [15, 43] cooperate with serial local spatial context model, which will cause slower decoding. Assuming similar spatial correlations in different slices, for the i-th slice, we first compute the attention map of decoded $i-1$-th slice, which is used to predict the global correlations in i-th slice. We also explore the global correlations in adjacent slices. Our approach for global spatial context modeling can work well with checkerboard-like patterns by attention masks. Finally, we fuse channel, local, global contexts, and side information to compute the distribution of latent representations. Our contributions are summarized as follows:

- We design multi-reference entropy models MEM and MEM$^+$ which combine local spatial, global spatial and channel contexts, and hyper-prior side information. Based on MEM and MEM$^+$, we propose MLIC and MLIC$^+$, which achieve state-of-the-art performance. We successfully explore the potential of an entropy model.
- To capture local spatial contexts, we design a stacked checkerboard context module and checkerboard attention to address the degradation of checkerboard context modeling while retaining two-pass decoding.
- We divide latent representation into slices and use the attention map of the previously decoded slice to predict the global correlations in the current slice. We also explore global correlations between adjacent slices.

2 RELATED WORKS

2.1 Learned Lossy Image Compression

According to rate-distortion optimization, large bit-rate $\mathcal{R}$ leads to lower distortion $\mathcal{D}$. It is a trade-off. Lagrange multiplier λ is used to adjust the weight of distortion to control the target bit-rate. The optimization target is

$$\mathcal{L} = \mathcal{R} + \lambda\mathcal{D}, \tag{1}$$

The basic learned image compression framework [4] consists analysis transform g_a, quantization Q, synthesis transform g_s and an entropy model to estimate rates. The process can be formulated as:

$$\boldsymbol{y} = g_a(\boldsymbol{x};\theta), \hat{\boldsymbol{y}} = Q(\boldsymbol{y}), \hat{\boldsymbol{x}} = g_s(\hat{\boldsymbol{y}};\phi), \tag{2}$$

where $\boldsymbol{x}$ is the input image, g_a transform the $\boldsymbol{x}$ to compact latent representation $\boldsymbol{y}$. $\boldsymbol{y}$ is quantized to $\hat{\boldsymbol{y}}$ for entropy coding. $\hat{\boldsymbol{x}}$ is the decompressed image. θ and ϕ are parameters of g_a and g_s. Quantization is non-differentiable, which can be addressed by adding uniform noise $\mathcal{U}(-0.5, 0.5)$ [4] or straight through estimator [46] during training. GDN/IGDN [3] layers are used to improve non-linearity. In the basic model, a factorized or a non-adaptive density entropy model is adopted. The estimated rate of $\hat{\boldsymbol{y}}$ is $\mathbb{E}[-\log_2 p_{\hat{\boldsymbol{y}}}(\hat{\boldsymbol{y}})]$.

A hyper-prior model is first introduced in [5], which extracts side information $\hat{\boldsymbol{z}}$ from $\boldsymbol{y}$. Hyper-prior model estimate distribution of $\hat{\boldsymbol{y}}$ from $\hat{\boldsymbol{z}}$. The rate of $\hat{\boldsymbol{y}}$ is $\mathbb{E}[-\log_2 p_{\hat{\boldsymbol{y}}|\hat{\boldsymbol{z}}}(\hat{\boldsymbol{y}}|\hat{\boldsymbol{z}})]$ and a uni-variate Gaussian distribution model for the hyper-prior is used. Some works extend it to a mean and scale Gaussian distribution [38], asymmetric Gaussian distribution [13] and Gaussian mixture model [12, 32] for more flexible distribution modeling.

2.2 Context-based Entropy Model

Many works [38, 39] have been proposed for more accurate context modeling, including local spatial, global spatial, and channel-wise context models.

Local spatial context models capture correlations between adjacent symbols. In [38], a pixel-cnn-like [52] masked convolutional layer is used to capture local correlations between $\hat{\boldsymbol{y}}_i$ and symbols $\hat{\boldsymbol{y}}_{<i}$, which leads to serial decoding. He *et al.* [18] divide latent representation $\hat{\boldsymbol{y}}$ into two parts $\hat{\boldsymbol{y}}_a$ and $\hat{\boldsymbol{y}}_{na}$ and use a checkerboard convolution to extract contexts of $\hat{\boldsymbol{y}}_{na}$ from $\hat{\boldsymbol{y}}_a$, achieving two-pass parallel decoding.

Some work aims to model correlations between distant symbols. In [43], neighbouring left and top symbols are used as bases for computing the similarity between the target symbol and its previous symbols. In [15], the latent representation is divided into 2 parts, the $L2$ distances of symbols in the first part are used to predict distant correlations in the second part. In [23], the side information is divided into global side information and local side information which leads to extra bits. However, these global context models are incorporated with the serial autoregressive context model, which further increases decoding latency.

Minnen et al. [39] model contexts between channels. $\hat{\boldsymbol{y}}$ is evenly divided to slices. The current slice $\hat{\boldsymbol{y}}^i$ is conditioned on previously decoded slices $\hat{\boldsymbol{y}}^{<i}$. An unevenly grouped channel-wise context model is proposed in [17] to address the uneven distribution of information among slices.

While some local and channel-wise context models [17, 35] have demonstrated impressive performance, effectively capturing local, global, and channel-wise contexts within a single entropy model remains a challenge. Addressing these correlations has the potential to further enhance the performance of the model.

3 METHOD

3.1 Motivation

According to information theory, conditional entropy is less than or equal to the entropy.

$$\mathbb{E}[-\log_2 p_{\hat{\boldsymbol{y}}}(\hat{\boldsymbol{y}})] \geq \mathbb{E}[-\log_2 p_{\hat{\boldsymbol{y}}|\boldsymbol{ctx}}(\hat{\boldsymbol{y}}|\boldsymbol{ctx})], \tag{3}$$

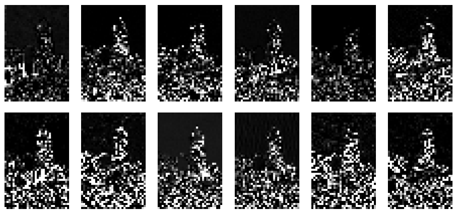

Figure 1: Visualization of channels of latent representation of Kodim19 extracted by Cheng'20 [12](optimized for MSE, $\lambda = 0.0483$).

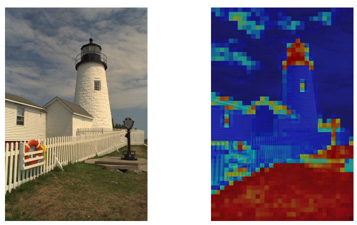

Figure 2: Heatmap of spatial cosine similarity of latent representation of Kodim19 extracted by Cheng'20 [12] (optimized for MSE, $\lambda = 0.0483$).

Notations	Explanation
$\boldsymbol{x}$, $\hat{\boldsymbol{x}}$	Input and reconstructed images
$\boldsymbol{y}$, $\hat{\boldsymbol{y}}$	Latent presentation and quantized latent representation
$\hat{\boldsymbol{y}}^i$, $\hat{\boldsymbol{y}}_a$, $\hat{\boldsymbol{y}}_{na}$	The i-th slice of $\hat{\boldsymbol{y}}$, anchor and non-anchor part of $\hat{\boldsymbol{y}}$
$\boldsymbol{z}$, $\hat{\boldsymbol{z}}$	Side information and quantized side information
g_{ep}, μ, σ	Entropy parameter module, Mean and scale of $\hat{\boldsymbol{y}}$
g_a, g_s, h_a, h_s	Analysis and synthesis transform, hyper analysis and synthesis
g_{ch}, g_{lc}, g_{gc}	Channel-wise, local spatial, and global spatial context modules
$g_{lc,stk}$	Stacked checkerboard context module
$g_{lc,attn}$	Shifted Window-based Checkerboard Attention
$g_{gc,intra}, g_{gc,inter}$	Intra-slice and Inter-slice global spatial context model
$\Phi_h, \Phi_{ch}, \Phi_{lc}, \Phi_{gc}$	hyperprior, channel-wise local spatial, and global spatial context
MEM ($^+$)	Multi-reference entropy model ($^+$)
M, N, S, K	Channel number of $\boldsymbol{y}$, $\boldsymbol{z}$, and $\hat{\boldsymbol{y}}^i$, kernel size
Q, AE, AD	Quantization, arithmetic encoding and decoing

Table 1: Explanations of notations.

	N	M	S	K	Entropy Model
MLIC	192	192	32	5	MEM ($g_{lc,stk}, g_{ch}, g_{gc,intra}$)
MLIC$^+$	192	320	32	5	MEM$^+$ ($g_{lc,attn}, g_{ch}, g_{gc,intra}, g_{gc,inter}$)

Table 2: Settings of MLIC, MLIC$^+$, MEM, and MEM$^+$.

where $\boldsymbol{ctx}$ is the context of $\hat{\boldsymbol{y}}$. As long as there are correlations in the latent representation $\hat{\boldsymbol{y}}$, exploiting these correlations can lead to bit savings.

In Figure 1 and Figure 2, we first illustrate channel-wise correlations and spatial correlations in latent representation of Kodim19 extracted by Cheng'20 [12]. In Figure 1, we visualize the feature of several channels. It's obvious that these features are quite similar. Capturing such correlations can be challenging for a spatial context model, since it employs the same mask for all channels when extracting contexts. This implies that certain correlations may not be fully captured. In Figure 2, symbols with the same color are of high correlation. A global context model is necessary to capture the correlation between symbols in the bottom-left corner and those in the bottom-right corner, where the grass features share similarities. Although a global context can capture local correlations, a global context can degrade to a local context model, which make it hard to capture distant correlations because of high similarity among adjacent symbols. Therefore, we argue that a local context model is necessary. The latent representation is with redundancy, which means there is potential to save bits by modeling such correlations. However, previous entropy models cannot model such correlations both in the spatial and channel domain. For a spatial context model, the interactions between channels are limited and for a channel-wise context model, there is no interaction in the current slice, which inspires us to design multi-reference entropy models. Our multi-reference entropy model capture correlations in the spatial and channel domain, which are introduced in the following sections.

3.2 Overview

3.2.1 MLIC and MLIC$^+$. We first give an overview of proposed models MLIC and MLIC$^+$. The architecture of MLIC and MLIC$^+$ is illustrated in Figure 3. MLIC and MLIC$^+$ share the same analysis transform g_a, synthesis transform g_s, hyper analysis h_a and hyper synthesis h_s, which are simplified from Cheng'20 [12]. We remove attention modules to reduce complexity. The difference between MLIC and MLIC$^+$ is the entropy model. MLIC is equipped with a Multi-reference entropy model MEM to balance the performance and complexity. MLIC$^+$ is equipped with a Multi-reference entropy model MEM$^+$ for better rate-distortion performance. The hyper-parameters and settings of MLIC and MLIC$^+$ are shown in Table 2. Same to Minnen et al. [39], we adopt *mixed quantization*, meaning adding uniform noise for entropy estimation, and adopting STE [46] to make quantization differentiable. Gaussian mean-scale distribution is adopted for entropy estimation.

3.2.2 MEM and MEM$^+$. The Multi-reference entropy models MEM and MEM$^+$ are able to capture channel-wise, local spatial, and global spatial correlations. To capture multi-correlations, our MEM and MEM$^+$ contain three parts: channel-wise context module g_{ch}, local spatial context module g_{lc} and global spatial context module g_{gc}. In the channel-wise context module, the latent representation $\hat{\boldsymbol{y}}$ is divided into slices $\{\hat{\boldsymbol{y}}^0, \hat{\boldsymbol{y}}^1, \cdots\}$. For the i-th slice $\hat{\boldsymbol{y}}^i$, the channel-wise context model captures the channel-wise context Φ^i_{ch} from slices $\hat{\boldsymbol{y}}^{<i}$. To capture local spatial correlations, we adopt the checkerboard pattern, where the latent representation $\hat{\boldsymbol{y}}$ is divided into anchor part $\hat{\boldsymbol{y}}_a$ and non-anchor part $\hat{\boldsymbol{y}}_{na}$. $\hat{\boldsymbol{y}}_a$ is local-context-free. We capture the local spatial context Φ_{lc} of $\hat{\boldsymbol{y}}_{na}$ from $\hat{\boldsymbol{y}}_a$. We propose two

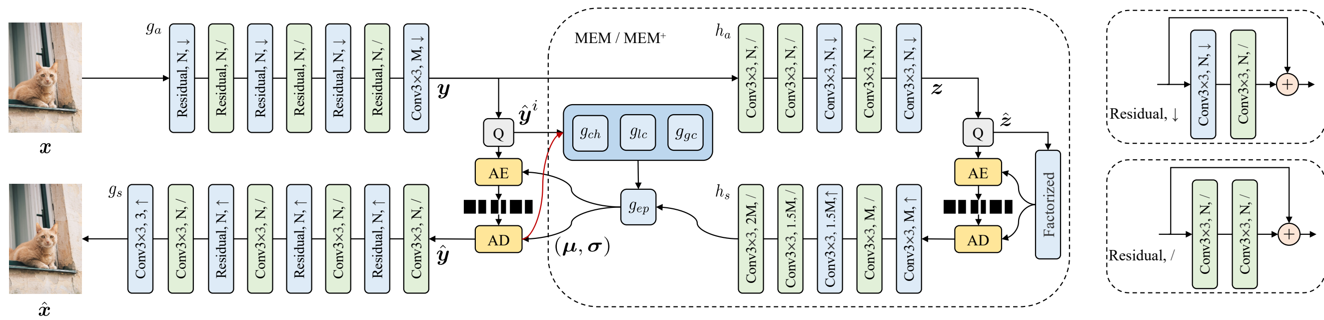

Figure 3: The overall architecture of MLIC and MLIC⁺. ↓ means down-sampling. ↑ means up-sampling. / means stride equals 1. Red line is the dataflow during decoding. Please refer to Table 1 for the explanations of other notations.

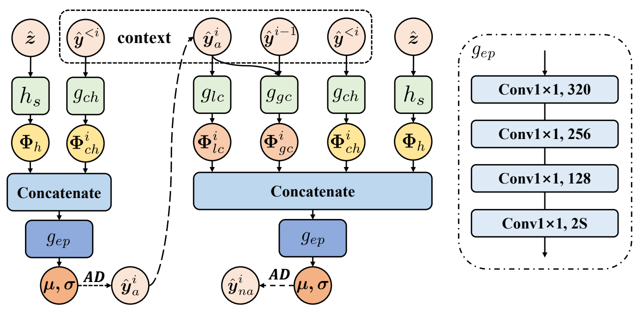

Figure 4: Multi-Reference Entropy Model MEM and MEM⁺. The figure illustrates the process of decoding a slice $\hat{\boldsymbol{y}}^i$.

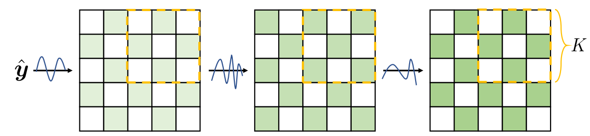

Figure 5: Stacked Checkerboard Context Module $g_{lc,stk}$. The figure illustrate changes of context information after every nonlinear transform. Green squares are context information.

different approaches: Stacked Checkerboard Context Module $g_{lc,stk}$ and Shifted Window-based Checkerboard Attention $g_{lc,attn}$ to capture local spatial contexts. We divide global spatial contexts $\boldsymbol{\Phi}_{gc}$ into intra-slice contexts $\boldsymbol{\Phi}_{gc,intra}$, and inter-slice contexts $\boldsymbol{\Phi}_{gc,inter}$. We propose Intra-Slice Global Context Module $g_{gc,intra}$ and Inter-Slice Global Context Module $g_{gc,inter}$ to capture such correlations. We introduce these modules in the following sections. The structures of MEM and MEM⁺ are shown in Table 2. MEM is not equipped with $g_{gc,inter}$ for less complexity. Following Minnen [39], Latent Residual Prediction modules [39] are adopted to cooperate with channel-wise context module g_{ch}. We illustrate the decompressing process of MLIC and MLIC⁺ in Figure 4. We use Equation 1 as our loss function and the estimated rate can be formulated as: $\mathcal{R} = \sum_{i=0}^{L} \mathcal{R}^i_a + \mathcal{R}^i_{na}$, where

$$\mathcal{R}^i_a = \mathbb{E}[-\log_2 p_{\hat{\boldsymbol{y}}^i_a|\Phi^i_{ch},\Phi_h}(\hat{\boldsymbol{y}}^i_a|\Phi^i_{ch}, \Phi_h)], \tag{4}$$

$$\mathcal{R}^i_{na} = \mathbb{E}[-\log_2 p_{\hat{\boldsymbol{y}}^i_{na}|\Phi^i_{ch},\Phi_h,\Phi^i_{lc},\Phi^i_{gc}}(\hat{\boldsymbol{y}}^i_{na}|\Phi^i_{ch}, \Phi_h, \Phi^i_{lc}, \Phi^i_{gc})]. \tag{5}$$

$\boldsymbol{\Phi}_h$ is the hyper-priors extracted by h_a and h_s.

3.3 Channel-wise Context Module

To extract channel-wise contexts, we first evenly divide latent representation $\hat{\boldsymbol{y}}$ into several slices $\{\hat{\boldsymbol{y}}^0, \hat{\boldsymbol{y}}^1, \cdots, \hat{\boldsymbol{y}}^L\}$, L is the number of slices. Slice $\hat{\boldsymbol{y}}^i$ is conditioned on slices $\hat{\boldsymbol{y}}^{<i}$. We use a channel context module g_{ch} to squeeze and extract context information from $\hat{\boldsymbol{y}}^{<i}$ when encoding and decoding $\hat{\boldsymbol{y}}^i$. g_{ch} consists of three 3×3 convolutional layers. The channel context becomes $\boldsymbol{\Phi}^i_{ch} = g_{ch}(\hat{\boldsymbol{y}}^{<i})$. The channel-wise context module g_{ch} helps select the most relative channels and extract information beneficial for accurate probability estimation. The channel number of each slice S is a hyper-parameter. Following Minnen et al [39], we set S to 32 in our models. We adopt latent residual prediction modules [39] to predict quantization error according to decoded slices and hyper-priors $\boldsymbol{\Phi}_h$.

3.4 Enhanced Checkerboard Context Module

The auto-regressive context model $g_{lc,ar}$ [38] leads serial decoding, while the checkerboard context model [18] makes parallel decoding possible. In the checkerboard context model, only half of the symbols are conditioned on decoded symbols, which leads to a slight degradation. We propose two different ways to solve it from different perspectives. Note that we capture local spatial contexts for each slice independently.

3.4.1 Stacked Checkerboard Context Module. Depth [19] and non-linearity are two important factors for boosting the performance of neural networks. The deeper and more non-linear the model is, the more expressiveness it has. In previous work [12, 18, 28, 38], local context module is a convolutional layer. In this module $g_{lc,stk}$, we stack J convolutional layers, which brings non-linearity and depth. According to the characteristics of the checkerboard pattern, we then point out that J should be an odd number. The odd-numbered convolution transfers the information extracted from the anchor part to the non-anchor part and the even-numbered convolution transfers the information extracted from the non-anchor part to the anchor part. The transfer process of context information is shown in Figure 5 when J is 3. We set $J = 3$ to balance model performance and parameters and set kernel size $K = 5$.

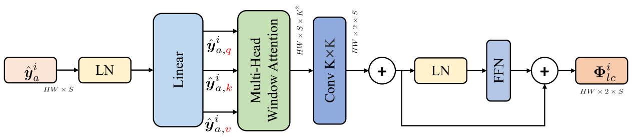

Figure 6: Checkerboard Attention Context Module $g_{lc,attn}$.

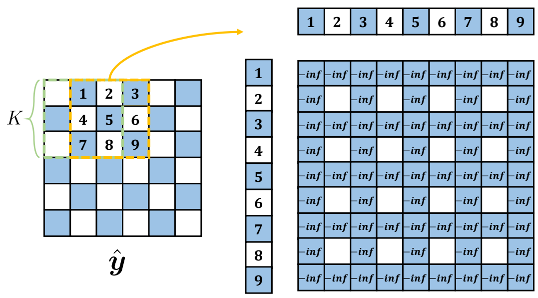

Figure 7: Mask of Shifted Window-based Checkerboard Attention $g_{lc,attn}$. Blue squares are non-anchor part $\hat{\boldsymbol{y}}_{na}$, white squares are anchor part $\hat{\boldsymbol{y}}_{a}$. Green square and yellow square are two windows.

Figure 8: Cosine similarity in the spatial domain of different slices of latent representation of Kodim19 extracted by Cheng'20 [12] (optimized for MSE).

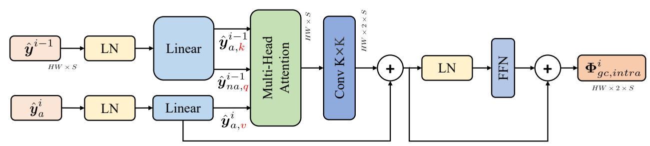

Figure 9: Intra-Slice Global Context Model $g_{gc,intra}$. S is the channel number of a slice.

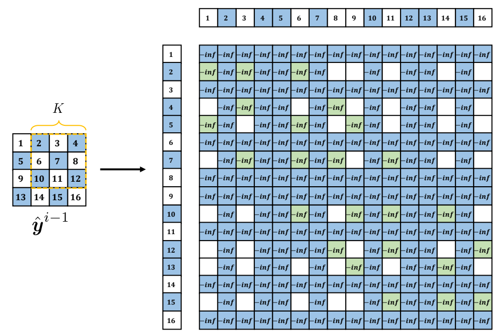

Figure 10: Mask of Intra-Slice Global Context Model $g_{gc,intra}$. Blue squares belong to non-anchor part $\hat{\boldsymbol{y}}_{na}^{i-1}$ and white squares belong to anchor part $\hat{\boldsymbol{y}}_{a}^{i-1}$ of slice $\hat{\boldsymbol{y}}^{i-1}$. The green squares are masked to avoid interactions in the local receptive field. The orange dotted box is the receptive field of the local context model.

3.4.2 Shifted-Window-based Checkerboard Attention. One drawback of the CNN-based local context module is the fixed weights, which makes them impossible to capture content-adaptive contexts. We contend that context-adaptation is beneficial due to the wide range of image diversity. In transformers [33, 53], the attention weight is generated dynamically according to the input, which inspires us to design a transformer-based content-adaptive local context module. The local receptive field is like a window, we capture the local spatial contexts by dividing the feature map into overlapped windows. We propose the checkerboard attention context module $g_{lc,attn}$. We take the i-th slice as example. Assumimg the resolution of the latent representation $\hat{\boldsymbol{y}}^i$ is $H \times W$, we divide $\hat{\boldsymbol{y}}^i$ into $H \times W$ overlapped windows and the window size is $K \times K$. To extract local correlations, we first compute the attention map of each window. Same as the convolutional checkerboard context model, interactions between $\boldsymbol{y}_a^i$ and $\boldsymbol{y}_{na}^i$ and interactions in $\boldsymbol{y}_{na}^i$ are not allowed. An example of the attention mask is illustrated in Figure 7. Such attention does not change the resolution of each window. We use a $K \times K$ convolutional layer to fusion local context information and make the size of the local context same as that of $\boldsymbol{y}^i$ before feeding it to an FFN [53]. The process be formulated as:

$$\hat{\boldsymbol{y}}_{attn}^i = \text{softmax}\left(\frac{\hat{\boldsymbol{y}}_{a,q}^i \times (\hat{\boldsymbol{y}}_{a,k}^i)^\top}{\sqrt{S}} + mask\right) \times \hat{\boldsymbol{y}}_{a,v}^i, \tag{6}$$

$$\hat{\boldsymbol{y}}_{conv}^i = \text{conv}_{K\times K}(\hat{\boldsymbol{y}}_{attn}^i), \tag{7}$$

$$\Phi_{lc}^i = \text{FFN}(\hat{\boldsymbol{y}}_{conv}^i) + \hat{\boldsymbol{y}}_{conv}^i, \tag{8}$$

where $\hat{\boldsymbol{y}}_{a,q}^i, \hat{\boldsymbol{y}}_{a,k}^i, \hat{\boldsymbol{y}}_{a,v}^i = \text{Embed}(\hat{\boldsymbol{y}}_a^i)$, $\hat{\boldsymbol{y}}_a^i$ is anchor part of i-th slice, *mask* the attention mask, S is the channel number of each slice.

Note that our overlapped window-partition is with linear complexity. The complexity of $g_{lc,attn}$ is $\Omega(2K^4HWS + 4HWS^2)$, where S is the channel number of a slice.

3.5 Global Spatial Context Module

We capture global-spatial contexts for each slice independently. We explore global correlations between non-anchor part $\hat{\boldsymbol{y}}_{na}^i$ and anchor part $\hat{\boldsymbol{y}}_a^i$ in a slice $\hat{\boldsymbol{y}}^i$ and between non-anchor part $\hat{\boldsymbol{y}}_{na}^i$ and slice $\hat{\boldsymbol{y}}^{i-1}$.

3.5.1 Intra-Slice Global Spatial Context Module. Due to codec consistency, it is impossible to know the global correlations between current symbols and other symbols during decoding. One solution is writing global correlations into bit-stream, which causes extra bits. In latent representation $\hat{\boldsymbol{y}}$, each channel contains different information but each channel can be treated as a thumbnail.

Methods	Kodak [25]		Tecnick [2]		CLIC Pro Val [48]		CLIC'21 Test [49]		CLIC'22 Test [50]		JPEGAI Test [22]	
	PSNR	MS-SSIM	PSNR	MS-SSIM	PSNR	MS-SSIM	PSNR	MS-SSIM	PSNR	MS-SSIM	PSNR	MS-SSIM
VTM-17.0 [8]	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Cheng'20 (CVPR'20) [12]	+5.58	−44.21	7.57	−39.61	+11.71	−41.29	+9.40	−37.22	+13.29	−33.40	+11.95	−40.03
Minnen'20 (ICIP'20) [39]	+3.23	−−	−0.88	−−	−−	−−	−−	−−	−−	−−	−−	−−
Qian'21 (ICLR'21) [43]	+10.05	−39.53	−−	−−	−−	−−	−−	−−	−−	−−	−−	−−
Xie'21 (MM'21) [57]	+1.55	−43.39	−0.80	−−	+3.21	−−	+0.99	−−	+2.13	−−	+2.35	−−
Entroformer (ICLR'22) [42]	+4.73	−42.64	+2.31	−−	−−	−−	−−	−−	−−	−−	−−	−−
SwinT-Charm (ICLR'22) [59]	−1.73	−42.64	+6.50	−−	−−	−−	+2.56	−−	−−	−−	+3.16	−−
WACNN (CVPR'22) [60]	−2.95	−47.71	−−	−−	+0.04	−44.38	−−	−−	−−	−−	−−	−−
STF (CVPR'22) [60]	−2.48	−47.72	−2.75	−−	+0.42	−44.82	−0.16	−−	+0.08	−−	+1.54	−−
ELIC (CVPR'22) [17]	−5.95	−44.60	−−	−−	−−	−−	−7.52	−−	−−	−−	−−	−−
NeuralSyntax (CVPR'22) [54]	+8.97	−39.56	−−	−−	+5.64	−38.92	−−	−−	−−	−−	−−	−−
Informer (CVPR'22) [23]	+10.01	−39.25	+9.72	−−	−−	−−	−−	−−	−−	−−	−−	−−
McQuic (CVPR'22) [58]	−1.57	−47.94	−−	−−	+6.82	−40.17	−−	−−	−−	−−	−−	−−
Contextformer (ECCV'22) [26]	−5.77	−46.12	−9.05	−42.29	−−	−−	−−	−−	−−	−−	−−	−−
Pan'22 (ECCV'22) [40]	+7.56	−36.20	+3.97	−−	−−	−−	−−	−−	−−	−−	−−	−−
MLIC (Ours)	**−8.05**	**−49.13**	**−12.73**	**−47.26**	**−8.79**	**−45.79**	**−11.17**	**−49.43**	**−10.89**	**−47.36**	**−9.90**	**−50.84**
MLIC^{+} (Ours)	**−11.39**	**−52.75**	**−16.38**	**−53.54**	**−12.56**	**−48.75**	**−15.03**	**−52.30**	**−14.85**	**−50.31**	**−13.42**	**−53.38**

Table 3: BD-Rate (%) comparison for PSNR (dB) and MS-SSIM (dB), with the best ones in red and second-best ones in blue. "−−" means the result is not available. The anchor is VTM-17.0 Intra.

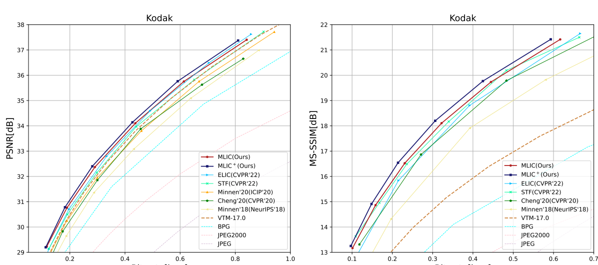

Figure 11: PSNR-Bit-rate curve (opt.MSE) and MS-SSIM-Bit-rate curve (opt.MS-SSIM) on Kodak dataset.

We point out that channels share similar global similarities. We illustrate the cosine similarity of two slices of Cheng'20 [12] in Figure 8. Their global correlations are similar despite differences in magnitude. When decoding the current slice $\hat{\boldsymbol{y}}^i$, decoded slice $\hat{\boldsymbol{y}}^{i-1}$ helps estimate the global correlations in slice $\hat{\boldsymbol{y}}^i$. One problem is how to eatimate the global correlations. Cosine similarity may be helpful, however, it is fixed and may not be accurate for feature. We point out attention map is a good choice. The embedding layer is learnable, which make it flexible to adjust the method for global correlations estimation by changing queries, keys, and values. We take the $i-1$-th slice and the i-th as example. When compressing or decompressing $\hat{\boldsymbol{y}}^i$, we first compute the correlations between anchor part $\hat{\boldsymbol{y}}^{i-1}_a$ and non-anchor part $\hat{\boldsymbol{y}}^{i-1}_{na}$ of slice $\hat{\boldsymbol{y}}^{i-1}$, because the checkerboard local context model makes anchor visible when decoding non-anchor part. We multiply the anchor part of current slice $\hat{\boldsymbol{y}}^i_a$ with an attention map. Due to the local correlations, adjacent symbols have similar global correlations. We use a $K \times K$ convolutional layer to refine the attention map by aggregating global similarities of adjacent symbols. The process of this Intra-Slice Global Context $g_{gc,inter}$ can be formulated as:

$$\hat{\boldsymbol{y}}^i_{attn} = \mathrm{softmax}\left(\frac{\hat{\boldsymbol{y}}^{i-1}_{na,q} \times (\hat{\boldsymbol{y}}^{i-1}_{a,k})^\top}{\sqrt{S}} + mask\right) \times \hat{\boldsymbol{y}}^i_{a,v}, \tag{9}$$

$$\hat{\boldsymbol{y}}^i_{conv} = \mathrm{conv}_{K\times K}(\hat{\boldsymbol{y}}^i_{attn}) + \hat{\boldsymbol{y}}^i_{a,v}, \tag{10}$$

$$\Phi^i_{gc,intra} = \mathrm{FFN}(\hat{\boldsymbol{y}}^i_{conv}) + \hat{\boldsymbol{y}}^i_{conv}, \tag{11}$$

where $\hat{\boldsymbol{y}}^{i-1}_{na,q}, \hat{\boldsymbol{y}}^{i-1}_{a,k} = \mathrm{Embed}(\hat{\boldsymbol{y}}^{i-1})$, $\hat{\boldsymbol{y}}^i_{a,v} = \mathrm{Embed}(\hat{\boldsymbol{y}}^i_a)$. Note that interactions within $\hat{\boldsymbol{y}}^{i-1}_{na}$ and $\hat{\boldsymbol{y}}^{i-1}_a$ are masked. Local receptive fields are also masked. We point out that if we don't adopt the mask, the intra-slice global context model can degrade to local context model becuase high similarities between adjacent symbols. Local correlations may dominate the attention map. The *mask* is illustrated in Figure 10.

3.5.2 Inter-Slice Global Context Module. Because of the global correlations between slices, we extend the intra-slice global context to the inter-slice global context. We explore the correlations between $\hat{\boldsymbol{y}}^i_{na}$ and $\hat{\boldsymbol{y}}^{i-1}$ by using $\hat{\boldsymbol{y}}^i_a$ as an approximation of $\hat{\boldsymbol{y}}^i_{na}$. We only explore the global correlations between adjacent slices to control complexity. In inter-slice global context module, we also adopt the learnable attention map and a convolutional layer for refinement. A mask is adopted to aviod interactions between $\hat{\boldsymbol{y}}^i_{na}$ and $\hat{\boldsymbol{y}}^{i-1}$. The process of this Inter-Slice Global Context Model $g_{gc,inter}$ can be formulated as:

$$\hat{\boldsymbol{y}}^i_{attn} = \mathrm{softmax}\left(\frac{\hat{\boldsymbol{y}}^i_{a,q} \times (\hat{\boldsymbol{y}}^{i-1}_k)^\top}{\sqrt{S}} + mask\right) \times \hat{\boldsymbol{y}}^{i-1}_v, \tag{12}$$

Methods	Kodak [25]	
	Encoding Time (s)	Decoding Time (s)
VTM-17.0 [8]	104.9218	0.2354
Cheng'20 (CVPR'20) [12]	3.7082	8.6586
Minnen'20 (ICIP'20) [39]	0.2467	0.1298
Xie'21 (MM'21) [57]	4.0973	9.1609
Entroformer (ICLR'22) [42]	4.7682	85.9190
WACNN (CVPR'22) [60]	0.2400	0.1400
STF (CVPR'22) [60]	0.2594	0.1629
ELIC*(CVPR'22) [17]	0.2315	0.2057
MLIC (Ours)	0.2202	0.1699
MLIC$^+$ (Ours)	0.3095	0.2767

ELIC* is reimplemented by us because official ELIC is not open-sourced.

Table 4: Encoding time and decoding time results compared with recent works.

$$\hat{\boldsymbol{y}}^i_{conv} = \text{conv}_{K\times K}(\hat{\boldsymbol{y}}^i_{attn}) + \hat{\boldsymbol{y}}^{i-1}_v, \tag{13}$$

$$\Phi^i_{gc,inter} = \text{FFN}(\hat{\boldsymbol{y}}^i_{conv}) + \hat{\boldsymbol{y}}^i_{conv}, \tag{14}$$

where $\hat{\boldsymbol{y}}^{i-1}_k, \hat{\boldsymbol{y}}^{i-1}_v = \text{Embed}(\hat{\boldsymbol{y}}^{i-1})$, $\hat{\boldsymbol{y}}^i_{a,q} = \text{Embed}(\hat{\boldsymbol{y}}^i_a)$.

4 EXPERIMENTS

4.1 Settings

We select 2×10^5 images from COCO2017 [31], DIV2K [1], ImageNet [45] with a resolution larger than 480×480 as our training set. We train our model on a single Tesla V100 GPU with various configurations of the Lagrange multiplier λ for different quality presets. We use MSE and MS-SSIM as distortion metrics. Following the settings of CompressAI [6], we set $\lambda \in \{18, 35, 67, 130, 250, 483\} \times 10^{-4}$ for MSE and $\lambda \in \{2.40, 4.58, 8.73, 16.64, 31.73, 60.50\}$ for MS-SSIM [55]. We train each model with an Adam optimizer with $\beta_1 = 0.9, \beta_2 = 0.999$ and the batch size is 8. We train each model for 2M steps. The learning rate starts at 10^{-4} and drops to 3×10^{-5} at 1.5M steps, drops to 10^{-5} at 1.8M steps, and drops to 3×10^{-6} at 1.9M steps, drops to 10^{-6} at 1.95M steps. During training, we random crop images to 256×256 patches during the first 1.2M steps, and crop images to 448×448 patches during the rest steps due to the sparsity of intra-slice and inter-slice attention masks shown in Figure 10. Large patches are beneficial for models to learn global references.

4.2 Performance

We evaluate our models on rate-distortion performance and codec efficiency.

4.2.1 Rate-Distortion Performance. Figure 11 shows the rate-distortion performance on Kodak [25] dataset. We report bd-rate reduction in Table 3 on Kodak [25], Tecnick [2], CLIC Pro Val [48], CLIC'21 Test [49], CLIC'22 Test [50], and JPEGAI Test [22] datasets. We compare our MLIC and MLIC$^+$ with recent models [12, 17, 23, 26, 39, 40, 42, 43, 54, 57–60] and VTM-17.0 [8]. Our MLIC and MLIC$^+$ achieve state-of-the-art performance on these datasets when measured in PSNR and MS-SSIM. Our MLIC and MLIC$^+$ reduce BD-rate by 8.05% and 11.39% on Kodak dataset over VVC when measured in PSNR. Compared with Cheng'20 [12], our MLIC can achieve a maximum improvement of 0.5 $\sim$ 0.8dB in PSNR and achieve a maximum improvement of 0.6dB in MS-SSIM on Kodak, our MLIC$^+$ can achieve a maximum improvement of 0.8 $\sim$ 1.0dB in PSNR. Our MLIC and MLIC$^+$ adopt simplified analysis transform and synthesis transform of Cheng'20 [12], therefore, the improvement of model performance is attributed to our Multi-Reference Entropy Models. Our Multi-Reference Entropy Models can capture more contexts. The improvement also proves correlations exist in multiple dimensions since Cheng'20 [12] adopts an spatial autoregressive context model. Compared with ELIC [17], our MLIC$^+$ can be up to 0.4db higher at low bit rate and reduce BD-rate by 6.23% over ELIC [17].

4.2.2 Qualitative Results. Figure 12 illustrates the example of reconstructed Kodim07 of our MLIC, our MLIC$^+$, Entroformer [42], Xie'21 [57], Cheng'20 [12] and VTM-17.0 [8]. PSNR value of our reconstructed images are 1dB higher than image reconstructed by VTM-17.0. Our reconstructed images retain more details with lower bpp. In terms of visual quality, our MLIC and MLIC$^+$ have significant improvements compared to other models. We provide more qualitative results in our supplementary material.

4.2.3 Codec Efficiency Analysis. In MLIC and MLIC$^+$, Our local spatial and global spatial context models are parallel. Although we divide $\hat{\boldsymbol{y}}$ into slices, since MLIC has only 6 slices and MLIC$^+$ has only 10 slices and the resolution of each slice is small, the serial processing between slices does not add too much time. We compare our MLIC and MLIC$^+$ with other recent models [12, 17, 39, 42, 57, 60] on encoding time, and decoding time. We include the arithmetic coding time. We compare encoding and decoding time on Kodak [25]. Our MLIC can encode and decode quite fast when compared with other models. Slice the entropy model of MLIC$^+$ is more complex, it takes slightly longer time to encode and decode an image.

4.3 Ablation Studies

4.3.1 Settings. We conduct corresponding ablation studies and evaluate the contributions of proposed entropy models on Kodak [25]. Each model is optimized for MSE. We train each model for 1.2M steps. We set learning rate to 10^{-4} and batch size to 8. We crop images to 256×256 patches during ablation studies. The results and configure are shown in Table 5. The base model is MLIC *w/o* context modules.

4.3.2 Analysis of Channel-wise Context Module. Channel-wise context module leads to a significant improvement in performance, possible to refer to symbols in the same and close position in the previous slices. The effectiveness of channel-wise context module proves the redundancy among channels.

4.3.3 Analysis of Local Context Module. Vanilla checkerboard context module leads to slight performance degradation while allows for two-pass decoding. Stacked checkerboard context module increases the depth for non-linearity, which leads to more powerful expressiveness. In ablation studies, $g_{lc,stk}$ saves 1.81% more bit-rates compared to $g_{lc,ckbd}$. Checkerboard attention module performs much better. $g_{lc,attn}$ saves 4.87% more bit-rates compared

Figure 12: Visualization of the reconstructed Kodim07 from the Kodak dataset. The metrics are [bpp↓/PNSR↑]. We compare our MLIC and MLIC$^+$ with Cheng'20 [12], Xie'21 [57], Entroformer [42] and VTM-17.0 [8].

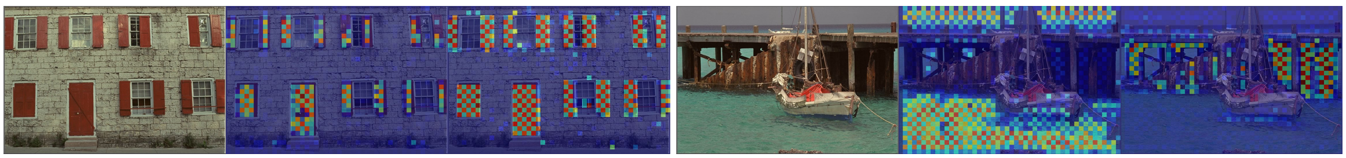

Figure 13: Attention map of Kodim01 and Kodim11 extracted by Intra-Global Context Model of MLIC (optimized for MSE, $\lambda = 0.0035$). Because interactions within anchor and non-anchor part are not allowed, the attention map is checkerboard-like.

	Kodak [25]
VTM-17.0 [8]	0.000
base + $g_{lc,ckbd}$	+14.82
base + $g_{lc,stk}$	+13.01
base + $g_{lc,attn}$	+9.95
base + g_{ch}	+2.73
base + $g_{lc,ckbd} + g_{ch}$	−0.98
base + $g_{lc,stk} + g_{ch}$	−2.23
base + $g_{lc,attn} + g_{ch}$	−2.92
base + $g_{ch} + g_{gc,intra}$	+1.24
base + $g_{ch} + g_{gc,intra,w/o\ \ mask}$	+1.91
base + $g_{lc,stk} + g_{ch} + g_{gc,intra}$	−4.11
base + $g_{lc,attn} + g_{ch} + g_{gc,intra}$	−4.90
base + $g_{lc,attn} + g_{ch} + g_{gc,intra} + g_{gc,inter}$	−5.63

Table 5: Ablation studies on Kodak [25]. The metric is BD-Rate (%) for PSNR (dB). The anchor is VTM-17.0 Intra.

to $g_{lc,ckbd}$, which can be attributed to the context-adaption and non-linearity of our proposed $g_{lc,attn}$.

4.3.4 Analysis of Global Context Module. We illustrate attention map of Intra-Slice Context Module in Figure 10. Our model successfully captures distant correlations, which are impossible for local context models to capture. Our Intra-Slice Global Context Module may be somewhat similar to the cross-attention model. However, we don't care about interactions between these two slices. We only use the attention map of $\hat{\boldsymbol{y}}^{i-1}$ to predict correlations in $\hat{\boldsymbol{y}}^{i}$. We also remove the mask in Intra-Slice Global Context Module $g_{gc,intra}$. Removing mask leads to performance degradation, because removing mask makes it hard for network to learn. When our proposed global context modules cooperate with local social context modules, the performance is further improved, which proves the necessarity of global spatial context modules for global correlation capturing and local spatial context modules for local correlation capturing. The gain of Inter-Slice Global Context Module is not very huge, which can be attributed to the approximation via anchor parts. One problem of our methods for global context is their quadratic computational complexity. One solution is cropping an image into patches. We find shared attention map and cropping an image into non-overlapped patches has almost no influence on performance. The results are reported in our supplementary material.

5 CONCLUSION

In this paper, we propose multi-reference entropy models MEM and MEM$^+$, which capture correlations in multiple dimensions. To our knowledge, this is the first successful attempt to capture channel, local and global correlations. Based on MEM and MEM$^+$, we obtain state-of-the-art models MLIC and MLIC$^+$. The significance of our work is investigating multiple correlations in latent representation and exploring the potential of an entropy model. However, due to the computational overhead, MLIC and MLIC$^+$ cannot be directly applied to mobile devices. We expect that this problem can be addressed by using knowledge distillation, network pruning, structural re-parameterization, and other light-weight designs.

ACKNOWLEDGMENTS

We thank the anonymous reviewers for their valuable comments and suggestions. This work is financially supported by National Natural Science Foundation of China U21B2012 and 62072013, Shenzhen Science and Technology Program-Shenzhen Cultivation of Excellent Scientific and Technological Innovation Talents project(Grant No. RCJC20200714114435057) , Shenzhen Science and Technology Program-Shenzhen Hong Kong joint funding project (Grant No. SGDX20211123144400001), this work is also financially supported for Outstanding Talents Training Fund in Shenzhen.

REFERENCES

[1] Eirikur Agustsson and Radu Timofte. 2017. NTIRE 2017 Challenge on Single Image Super-Resolution: Dataset and Study. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.

[2] Nicola Asuni and Andrea Giachetti. 2014. TESTIMAGES: a Large-scale Archive for Testing Visual Devices and Basic Image Processing Algorithms.. In *STAG*. 63–70.

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

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

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

[6] Jean Bégaint, Fabien Racapé, Simon Feltman, and Akshay Pushparaja. 2020. CompressAI: a PyTorch library and evaluation platform for end-to-end compression research. *arXiv preprint arXiv:2011.03029* (2020).

[7] Fabrice Bellard. 2015. BPG Image format. https://bellard.org/bpg.

[8] Benjamin Bross, Ye-Kui Wang, Yan Ye, Shan Liu, Jianle Chen, Gary J. Sullivan, and Jens-Rainer Ohm. 2021. Overview of the Versatile Video Coding (VVC) Standard and its Applications. *IEEE Transactions on Circuits and Systems for Video Technology* 31, 10 (2021), 3736–3764. https://doi.org/10.1109/TCSVT.2021.3101953

[9] Maryline Charrier, Diego Santa Cruz, and Mathias Larsson. 1999. JPEG2000, the Next Millennium Compression Standard for Still Images. In *IEEE International Conference on Multimedia Computing and Systems, ICMCS 1999, Florence, Italy, June 7-11, 1999. Volume I*. IEEE Computer Society, 131–132. https://doi.org/10.1109/MMCS.1999.779134

[10] Fangdong Chen, Yumeng Xu, and Li Wang. 2022. Two-Stage Octave Residual Network for End-to-end Image Compression. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 3922–3929.

[11] Tong Chen, Haojie Liu, Zhan Ma, Qiu Shen, Xun Cao, and Yao Wang. 2021. End-to-End Learnt Image Compression via Non-Local Attention Optimization and Improved Context Modeling. *IEEE Transactions on Image Processing* 30 (2021), 3179–3191.

[12] Zhengxue Cheng, Heming Sun, Masaru Takeuchi, and Jiro Katto. 2020. Learned Image Compression With Discretized Gaussian Mixture Likelihoods and Attention Modules. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[13] Ze Cui, Jing Wang, Shangyin Gao, Tiansheng Guo, Yihui Feng, and Bo Bai. 2021. Asymmetric Gained Deep Image Compression With Continuous Rate Adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 10532–10541.

[14] Ge Gao, Pei You, Rong Pan, Shunyuan Han, Yuanyuan Zhang, Yuchao Dai, and Hojae Lee. 2021. Neural Image Compression via Attentional Multi-Scale Back Projection and Frequency Decomposition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 14677–14686.

[15] Zongyu Guo, Zhizheng Zhang, Runsen Feng, and Zhibo Chen. 2021. Causal Contextual Prediction for Learned Image Compression. *IEEE Transactions on Circuits and Systems for Video Technology* (2021).

[16] Zongyu Guo, Zhizheng Zhang, Runsen Feng, and Zhibo Chen. 2021. Soft then hard: Rethinking the quantization in neural image compression. In *International Conference on Machine Learning*. PMLR, 3920–3929.

[17] Dailan He, Ziming Yang, Weikun Peng, Rui Ma, Hongwei Qin, and Yan Wang. 2022. ELIC: Efficient Learned Image Compression With Unevenly Grouped Space-Channel Contextual Adaptive Coding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 5718–5727.

[18] Dailan He, Yaoyan Zheng, Baocheng Sun, Yan Wang, and Hongwei Qin. 2021. Checkerboard Context Model for Efficient Learned Image Compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 14771–14780.

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[20] Yueyu Hu, Wenhan Yang, and Jiaying Liu. 2020. Coarse-to-fine hyper-prior modeling for learned image compression. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 11013–11020.

[21] Nick Johnston, Damien Vincent, David Minnen, Michele Covell, Saurabh Singh, Troy Chinen, Sung Jin Hwang, Joel Shor, and George Toderici. 2018. Improved Lossy Image Compression With Priming and Spatially Adaptive Bit Rates for Recurrent Networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[22] JPEG-AI. 2020. JPEG-AI Test Images. https://jpegai.github.io/test_images/.

[23] Jun-Hyuk Kim, Byeongho Heo, and Jong-Seok Lee. 2022. Joint Global and Local Hierarchical Priors for Learned Image Compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 5992–6001.

[24] Diederik P. Kingma and Max Welling. 2014. Auto-Encoding Variational Bayes. In *International Conference on Learning Representations*.

[25] Eastman Kodak. 1993. Kodak lossless true color image suite (PhotoCD PCD0992). http://r0k.us/graphics/kodak.

[26] A Burakhan Koyuncu, Han Gao, and Eckehard Steinbach. 2022. Contextformer: A Transformer with Spatio-Channel Attention for Context Modeling in Learned Image Compression. *arXiv preprint arXiv:2203.02452* (2022).

[27] Jooyoung Lee, Seunghyun Cho, and Seung-Kwon Beack. 2019. Context-adaptive entropy model for end-to-end optimized image compression. In *International Conference on Learning Representations*.

[28] Mu Li, Kede Ma, Jane You, David Zhang, and Wangmeng Zuo. 2020. Efficient and Effective Context-Based Convolutional Entropy Modeling for Image Compression. *IEEE Transactions on Image Processing* 29 (2020), 5900–5911.

[29] Mu Li, Wangmeng Zuo, Shuhang Gu, Debin Zhao, and David Zhang. 2018. Learning convolutional networks for content-weighted image compression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3214–3223.

[30] Chaoyi Lin, Jiabao Yao, Fangdong Chen, and Li Wang. 2020. A Spatial RNN Codec for End-to-End Image Compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 13269–13277.

[31] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*. Springer, 740–755.

[32] Jiaheng Liu, Guo Lu, Zhihao Hu, and Dong Xu. 2020. A unified end-to-end framework for efficient deep image compression. *arXiv preprint arXiv:2002.03370* (2020).

[33] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. 2021. Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 10012–10022.

[34] Ming Lu, Peiyao Guo, Huiqing Shi, Chuntong Cao, and Zhan Ma. 2022. Transformer-based Image Compression. In *Data Compression Conference (DCC)*. 469–469. https://doi.org/10.1109/DCC52660.2022.00080

[35] Changyue Ma, Zhao Wang, Ruling Liao, and Yan Ye. 2021. A Cross Channel Context Model for Latents in Deep Image Compression. *arXiv preprint arXiv:2103.02884* (2021).

[36] Haichuan Ma, Dong Liu, Ning Yan, Houqiang Li, and Feng Wu. 2020. End-to-end optimized versatile image compression with wavelet-like transform. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2020).

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

[38] David Minnen, Johannes Ballé, and George D Toderici. 2018. Joint autoregressive and hierarchical priors for learned image compression. In *Advances in Neural Information Processing Systems*. 10771–10780.

[39] David Minnen and Saurabh Singh. 2020. Channel-wise autoregressive entropy models for learned image compression. In *2020 IEEE International Conference on Image Processing (ICIP)*. IEEE, 3339–3343.

[40] Guanbo Pan, Guo Lu, Zhihao Hu, and Dong Xu. 2022. Content Adaptive Latents and Decoder for Neural Image Compression. In *European Conference on Computer Vision*. Springer, 556–573.

[41] William B Pennebaker and Joan L Mitchell. 1992. *JPEG: Still image data compression standard*. Springer Science & Business Media.

[42] Yichen Qian, Ming Lin, Xiuyu Sun, Zhiyu Tan, and Rong Jin. 2022. Entroformer: A Transformer-based Entropy Model for Learned Image Compression. In *International Conference on Learning Representations*.

[43] Yichen Qian, Zhiyu Tan, Xiuyu Sun, Ming Lin, Dongyang Li, Zhenhong Sun, Hao Li, and Rong Jin. 2021. Learning Accurate Entropy Model with Global Reference for Image Compression. In *International Conference on Learning Representations*.

[44] Oren Rippel and Lubomir Bourdev. 2017. Real-time adaptive image compression. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 2922–2930.

[45] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. 2015. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)* 115, 3 (2015), 211–252. https://doi.org/10.1007/s11263-015-0816-y

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

[47] George Toderici, Sean M. O'Malley, Sung Jin Hwang, Damien Vincent, David Minnen, Shumeet Baluja, Michele Covell, and Rahul Sukthankar. 2016. Variable Rate Image Compression with Recurrent Neural Networks. In *International Conference on Learning Representations*.

[48] George Toderici, Radu Timofte, Johannes Ballé, Eirikur Agustsson, Nick Johnston, and Fabian Mentzer. 2020. Workshop and Challenge on Learned Image Compression (CLIC). http://www.compression.cc.

[49] George Toderici, Radu Timofte, Johannes Ballé, Eirikur Agustsson, Nick Johnston, and Fabian Mentzer. 2021. Workshop and Challenge on Learned Image Compression (CLIC). clic.compression.cc/2021/.

[50] George Toderici, Radu Timofte, Johannes Ballé, Eirikur Agustsson, Nick Johnston, and Fabian Mentzer. 2022. Workshop and Challenge on Learned Image

Compression (CLIC). http://clic.compression.cc.

[51] George Toderici, Damien Vincent, Nick Johnston, Sung Jin Hwang, David Minnen, Joel Shor, and Michele Covell. 2017. Full Resolution Image Compression With Recurrent Neural Networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[52] Aäron Van Den Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. 2016. Pixel recurrent neural networks. In *International conference on machine learning*. PMLR, 1747–1756.

[53] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017), 5998–6008.

[54] Dezhao Wang, Wenhan Yang, Yueyu Hu, and Jiaying Liu. 2022. Neural Data-Dependent Transform for Learned Image Compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 17379–17388.

[55] Zhou Wang, Eero P Simoncelli, and Alan C Bovik. 2003. Multiscale structural similarity for image quality assessment. In *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, Vol. 2. Ieee, 1398–1402.

[56] Yaojun Wu, Xin Li, Zhizheng Zhang, Xin Jin, and Zhibo Chen. 2021. Learned block-based hybrid image compression. *IEEE Transactions on Circuits and Systems for Video Technology* (2021).

[57] Yueqi Xie, Ka Leong Cheng, and Qifeng Chen. 2021. Enhanced Invertible Encoding for Learned Image Compression. In *Proceedings of the ACM International Conference on Multimedia*.

[58] Xiaosu Zhu, Jingkuan Song, Lianli Gao, Feng Zheng, and Heng Tao Shen. 2022. Unified Multivariate Gaussian Mixture for Efficient Neural Image Compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 17612–17621.

[59] Yinhao Zhu, Yang Yang, and Taco Cohen. 2022. Transformer-based Transform Coding. In *International Conference on Learning Representations*.

[60] Renjie Zou, Chunfeng Song, and Zhaoxiang Zhang. 2022. The Devil Is in the Details: Window-Based Attention for Image Compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 17492–17501.