Diversify, Contextualize, and Adapt: Efficient Entropy Modeling for Neural Image Codec

Jun-Hyuk Kim* **Seungeon Kim** **Won-Hee Lee** **Dokwan Oh**
Samsung Advanced Institute of Technology
{jh131.kim, se2.kim, why_wh.lee, dokwan.oh}@samsung.com

Abstract

Designing a fast and effective entropy model is challenging but essential for practical application of neural codecs. Beyond spatial autoregressive entropy models, more efficient backward adaptation-based entropy models have been recently developed. They not only reduce decoding time by using smaller number of modeling steps but also maintain or even improve rate–distortion performance by leveraging more diverse contexts for backward adaptation. Despite their significant progress, we argue that their performance has been limited by the simple adoption of the design convention for forward adaptation: using only a single type of hyper latent representation, which does not provide sufficient contextual information, especially in the first modeling step. In this paper, we propose a simple yet effective entropy modeling framework that leverages sufficient contexts for forward adaptation without compromising on bit-rate. Specifically, we introduce a strategy of diversifying hyper latent representations for forward adaptation, i.e., using two additional types of contexts along with the existing single type of context. In addition, we present a method to effectively use the diverse contexts for contextualizing the current elements to be encoded/decoded. By addressing the limitation of the previous approach, our proposed framework leads to significant performance improvements. Experimental results on popular datasets show that our proposed framework consistently improves rate–distortion performance across various bit-rate regions, e.g., 3.73% BD-rate gain over the state-of-the-art baseline on the Kodak dataset.

1 Introduction

Most neural image codecs [8, 9, 11, 15, 17, 18] first transform an image into a quantized latent representation. It is then encoded into a bitstream via an entropy coding algorithm, which relies on a learned probability model known as the entropy model. According to the Shannon's source coding theorem, the minimum expected length of a bitstream is equal to the entropy of the source. Thus, accurately modeling entropy of the quantized latent representation is crucial.

Entropy models estimate a joint probability distribution over the elements of the quantized latent representation. Generally, it is assumed that all elements follow conditionally independent probability distributions. To satisfy this, the probability distributions are modeled in context-adaptive manners, which is key to accurate entropy modeling [18]. Recent methods are based on the joint backward and forward adaptation where the probability distributions adapt by leveraging contexts in two different ways: directly using previously encoded/decoded elements (i.e., backward adaptation), and extracting and utilizing an additional hyper latent representation (i.e., forward adaptation). Here, the type of contexts leveraged can be diverse depending on the spatial range they cover. First, each element has dependencies with other elements in the same spatial location along the channel dimension. Since the channel-wise dependencies correspond to the local image area (e.g., a 16×16 patch), we denote

*Corresponding author

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

 https://doi.org/10.52202/079017-1461

them as the "local" context. Second, dependencies exist among spatially adjacent elements, and we refer to them as the "regional" context. Lastly, long-range spatial dependencies span the entire image area, referred to as the "global" context.

For the backward adaptation, the modeling order, i.e., which elements are modeled first, is an important factor, and the key lies in how effectively we can utilize diverse contexts in the modeling process. Early studies employ spatial autoregressive (AR) models that access regional context including the most spatially adjacent elements. However, they suffer from significantly slow decoding times due to the inevitably large number of modeling steps, which is equal to the spatial dimensions [18]. To enhance efficiency in entropy modeling, several attempts reduce the number of modeling steps while leveraging diverse contexts: a 10-step channel-wise AR model [17], a 2-step spatial non-AR model with a checkerboard pattern [8], and a 4-step non-AR model that operates across spatial and channel dimensions using a quadtree partition [14].

Entropy models based on the efficient backward adaptation methods have led to significant improvements. However, they are still limited in fully leveraging contexts for forward adaptation. Since they use multiple neural layers with downsampling and upsampling for modeling hyper latent representation, they can only access the regional context. This limits the performance improvement due to the insufficient contexts (Figure 1a). In particular, this limitation is exacerbated at the first step where only forward adaptation is utilized due to the absence of previous elements (Figure 6). Therefore, it is necessary to develop effective forward adaptation in synergy with the efficient backward adaptation.

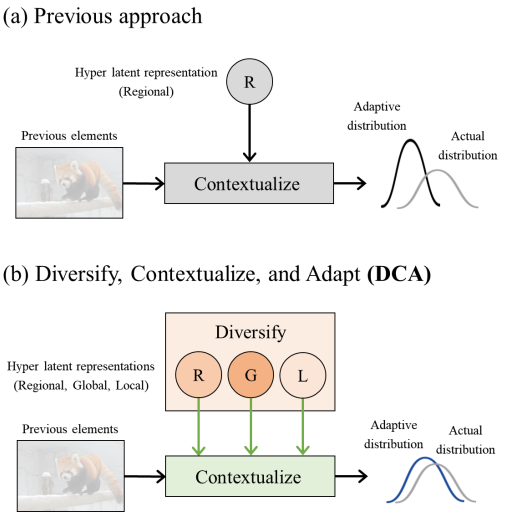


Figure 1: **DCA diversifies** the hyper latent representations and **contextualizes** the current elements by leveraging the diverse hyper latent representations along with the previous elements. As a result, the probability distributions **adapt** effectively, leading to accurate entropy modeling.

In this paper, we propose a simple yet effective entropy modeling framework, called **DCA** (Diversify, Contextualize, and Adapt), leveraging sufficient contexts for forward adaptation without compromising on bit-rate (Figure 1b). Building on the quadtree partition-based backward adaptation [14], we introduce a strategy of diversification, i.e., extracting local, regional, and global hyper latent representations unlike only a single regional one in the previous approach. Note that simply using more contexts for forward adaptation does not guarantee performance improvements because forward adaptation requires additional bit allocation unlike backward adaptation. Then, we propose how to effectively utilize the diverse contexts along with the previously modeled elements for contextualizing the current elements to be encoded/decoded. To consider stepwise different situations, e.g., increased number of previous elements over steps, our contextualization method is designed to utilize each hyper latent representation separately in a step-adaptive manner. Additionally, our contextualization method proceeds in the sequence of regional, global, and local hyper latent representations. Similarly to backward adaptation, we empirically observe that modeling order also matters in forward adaptation.

Our main contributions are summarized as follows:

- We propose a strategy of diversifying contexts for forward adaptation by extracting three different hyper latent representations, i.e., local, regional, and global ones. This strategy can provide sufficient contexts for forward adaptation without compromising on bit-rate.
- We introduce how to effectively leverage the diverse contexts, i.e., previously modeled elements and the three hyper latent representations. We empirically show that the modeling order of three types of contexts affects the performance.
- Through the diversification and contextualization methods, our DCA effectively adapts, resulting in significant performance improvements. For example, DCA achieves 3.73% BD-rate gain over the state-of-the-art method [14] on the Kodak dataset.

https://doi.org/10.52202/079017-1461

2 Related work

Joint backward and forward adaptation. Ballé et al. [3] propose a scale hyperprior for forward adaptation. A hyper latent representation is extracted and utilized for inferring local scale parameters of the parameterized entropy model. Minnen et al. [18] extend the hyperprior model by using an additional mean hyperprior, and introduce joint backward and forward adaptation by combining the extended hyperprior model with a spatial autoregressive (AR) model. A patch matching-based non-local referring model [20] and a multi-head attention-based global hyperprior [11] are proposed to enrich contexts for backward and forward adaptation, respectively.

Efficient backward adaptation. To address the slow decoding times of spatial AR-based entropy models, several studies have proposed group-wise backward adaptation methods. They first divide the quantized latent representation into multiple groups and then process them in a group-wise manner, resulting in improved efficiency. He et al. [8] propose dividing the quantized latent representation into two groups using the checkerboard pattern, which is further improved by incorporating Transformer-based modules [21]. While they apply a group-wise modeling in spatial dimension, Minnen and Singh [17] introduce a channel-wise AR model that divides the quantized latent representation into ten groups along channel dimension. Some studies [22, 23] improve this model by applying Swin Transformer [16]. Based on the channel-wise AR model, He et al. [9] optimize the channel division and combine it with the checkerboard-based model. Recently, Li et al. [14] propose a quadtree partition-based backward adaptation that divides the quantized latent representation into four groups considering both channel and spatial dimensions.

In this paper, we propose a novel fast and effective entropy model that achieves better rate–distortion performance by diversifying not only the quantized latent representation but also the hyper latent representations for backward and forward adaptation, respectively.

3 Methods

We provide an overview of the proposed methods in Figure 2. The analysis transform $f_a(\cdot)$ and the synthesis transform $f_s(\cdot)$ (the gray blocks in Figure 2) are learned to find an effective mapping between an input image $\boldsymbol{x}$ and a quantized latent representation $\hat{\boldsymbol{y}}$, i.e., $\hat{\boldsymbol{y}} = \lfloor f_a(\boldsymbol{x}) \rceil$ and $\hat{\boldsymbol{x}} = f_s(\hat{\boldsymbol{y}})$, where $\lfloor \cdot \rceil$ is a round operation and $\hat{\boldsymbol{x}}$ is the decoded image. For the analysis and synthesis transforms, we adopt the same model structure as in the ELIC-sm model [9] due to its efficiency.

All other components are learned to model a prior probability distribution on the quantized latent representation $\hat{\boldsymbol{y}}$, i.e., the entropy model $p_{\hat{\boldsymbol{y}}}$. The learned entropy model is utilized in the process of entropy coding, for which we employ the asymmetric numeral systems [6]. Here, our goal is to design a fast and effective learned entropy model.

Quadtree partition. We build our entropy model on the joint forward and backward adaptation where the quadtree partition is used, which is formulated as follows [14]:

$$p(\hat{\boldsymbol{y}}) = p(\hat{\boldsymbol{z}}) \times p(\hat{\boldsymbol{y}}|\hat{\boldsymbol{z}}) = p(\hat{\boldsymbol{z}}) \times \prod_{i=1}^{4} p(\hat{\boldsymbol{y}}^i|\hat{\boldsymbol{y}}^{<i}, \hat{\boldsymbol{z}})$$

where $\hat{\boldsymbol{y}}$ is the quantized latent representation, $\hat{\boldsymbol{z}}$ is the quantized regional hyper latent representation, $\hat{\boldsymbol{y}}^i$ is the elements to be modeled at the i-th step, and $\hat{\boldsymbol{y}}^{<i}$ is all the previous modeled elements before the i-th step. At each step, one-fourth of the total elements are modeled. The method partitions the quantized latent representation into four groups along the channel dimension, and then divides each group into non-overlapping 2×2 patches along the spatial dimension. The entropy modeling proceeds over

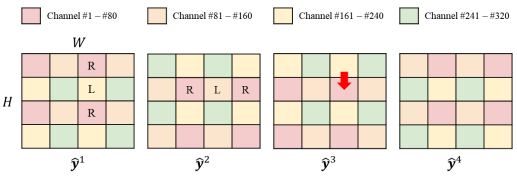


Figure 3: Example of the quadtree partition-based backward adaptation for $\hat{\boldsymbol{y}} \in \mathbb{R}^{4\times4\times320}$. For simplicity, channel dimensions are represented via different colors. $\hat{\boldsymbol{y}}^i$ means the elements to be encoded/decoded at the i-th step. For modeling the current elements $\hat{\boldsymbol{y}}^i$, all the previous modeled elements $\hat{\boldsymbol{y}}^{<i}$ are used. For example, the elements corresponding to the red arrow leverage diverse contexts including elements across different channels at the same spatial location (local context denoted as L) and spatially adjacent four elements of the same channel (regional context denoted as R).

https://doi.org/10.52202/079017-1461

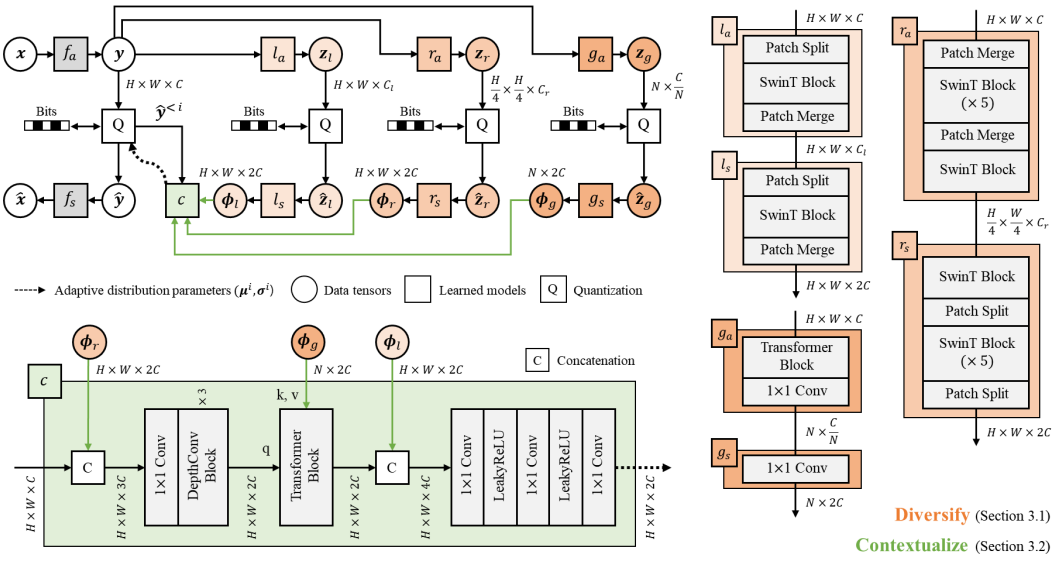


Figure 2: Overview of the neural image codec with the proposed entropy model, referred to as DCA. DCA can be employed by any analysis and synthesis transforms $f_a(\cdot)$ and $f_s(\cdot)$. DCA is an adaptive entropy model consisting of two main stages: **diversify** (Section 3.1) and **contextualize** (Section 3.2). First, given the latent representation $\boldsymbol{y}$, DCA extracts diverse hyper latent representations $\hat{\boldsymbol{z}}_l$, $\hat{\boldsymbol{z}}_r$, and $\hat{\boldsymbol{z}}_g$, and then encodes them into the bitstreams using learned factorized entropy models, which are omitted in this figure for simplicity. Second, contextualization proceeds over four steps. By using the three features $\boldsymbol{\phi}_l$, $\boldsymbol{\phi}_r$, and $\boldsymbol{\phi}_g$ (from the three hyper latent representations, respectively) and all the previously encoded/decoded elements before the i-th step, i.e., $\hat{\boldsymbol{y}}^{<i}$, DCA contextualizes the current elements to be encoded/decoded, i.e., $\hat{\boldsymbol{y}}^i$, and finally obtains adaptive distribution parameters $\boldsymbol{\mu}^i$ and $\boldsymbol{\sigma}^i$ for probability modeling. Using the learned adaptive probability model, the quantized latent representation $\hat{\boldsymbol{y}}$ are encoded into a bitstream.

four steps, with each step modeling different elements as shown in Figure 3. This quadtree partition-based method uses diverse contexts for backward adaptation, capturing dependencies from both spatial and channel dimensions.

Motivation. Recent studies to efficient modeling of backward adaptation have made significant advancements in terms of optimizing the rate–distortion–computation trade-off; however, there is still a gap between their assumptions in the probability modeling and actual data, leaving room for further performance enhancement. The assumptions are as follows: 1) All elements of $\hat{\boldsymbol{z}}$ are independent; 2) All elements of $\hat{\boldsymbol{y}}^i$ are conditionally independent given $\hat{\boldsymbol{y}}^{<i}$ and $\hat{\boldsymbol{z}}$. Here, the more the actual data deviates from the assumptions, the lower the accuracy of the entropy modeling. At the first modeling step, the elements are modeled conditioned only on the quantized hyper latent representation, i.e., $p(\hat{\boldsymbol{y}}^1|\hat{\boldsymbol{z}})$, resulting in a hyperprior model known for deviating from to the second assumption [18]. This can be more problematic because the state-of-the-art methods process a relatively large number of elements at the first step in order to complete the overall modeling with a minimal number of steps. We also empirically show that this problem actually occurs in Figure 6.

One straightforward solution is to increase the number of steps so that fewer elements are modeled in the first step. However, this leads to slower modeling speeds, which conflict with the goal of our paper, i.e., developing a fast and effective entropy model. Another simple approach is to provide more quantized regional hyper latent representation $\hat{\boldsymbol{z}}$ when modeling the quantized latent representation $\hat{\boldsymbol{y}}^1$. However, paradoxically, this approach can introduce another issue due to the first assumption. Since all elements of hyper latent representation are the same type of information (i.e., regional context), there is a relatively high likelihood of dependencies among the elements. Therefore, to meet both assumptions, the newly added hyper latent representation is required to be independent from the existing regional hyper latent representation. This is why our proposed diversification method using

local, regional, and global hyper latent representations is needed. In this paper, we propose a fast and effective entropy model, called DCA, which consists of three main stages: diversifying the hyper latent representations, contextualizing the elements targeted for probability modeling, and ultimately adapting the probability distribution of the elements to the given contexts.

3.1 Diversify

The proposed DCA aims to diversify the information that the hyper latent representations contain. Specifically, given the latent representation $\boldsymbol{y} \in \mathbb{R}^{H \times W \times C}$, where H, W, and C are the height, width, and the number of channels, respectively, DCA extracts three different types of hyper latent representations depending on the range they cover: a local hyper latent representation $\hat{\boldsymbol{z}}_l \in \mathbb{R}^{H \times W \times C_l}$, a regional hyper latent representation $\hat{\boldsymbol{z}}_r \in \mathbb{R}^{\frac{H}{4} \times \frac{W}{4} \times C_r}$, and a global hyper latent representation $\hat{\boldsymbol{z}}_g \in \mathbb{R}^{N \times \frac{C}{N}}$. The whole process is illustrated in the orange blocks of Figure 2.

Local context. To model remaining dependencies along channel dimension at each spatial location, which correspond to a 16×16 local patch in the image domain, we introduce local hyper analysis and synthesis transforms, $l_a(\cdot)$ and $l_s(\cdot)$, based on Swin Transformer (SwinT) [16]. The local hyper analysis transform $l_a(\cdot)$ analyzes local information in the latent representation, followed by the quantization operation to obtain a local hyper latent representation $\hat{\boldsymbol{z}}_l$. The local synthesis transform $l_s(\cdot)$ synthesizes the local features $\boldsymbol{\phi}_l \in \mathbb{R}^{H \times W \times 2C}$ for contextualization from the local hyper latent representation $\hat{\boldsymbol{z}}_l$.

Each transform proceeds in the order of a Patch Split block, a SwinT block, and a Patch Merge block. The Patch Split block serves the function of shifting all channel-wise elements at each spatial location to a 2×2 spatial resolution, consisting of the depth-to-space, layer normalization, and linear layers in sequence. The SwinT block then captures dependencies between elements within each non-overlapping window of the input, producing an output of the same size as the input. By setting the window size to 2×2 in conjunction with the use of the Split block, we enforce the local hyper transforms to focus only on the local image area. The Patch Merge block performs the opposite function of the Patch Split block, containing the layer normalization, linear, and space-to-depth layers in sequence.

Regional context. While the receptive field of the local hyper transforms is limited to the local image area (i.e., 16×16 patches), regional hyper analysis transform $r_a(\cdot)$ and regional hyper synthesis transform $r_s(\cdot)$ model remaining dependencies between elements distributed across a relatively wide image area. The regional hyper analysis transform $r_a(\cdot)$ analyzes regional information in the latent representation and yields a regional hyper latent representation $\hat{\boldsymbol{z}}_r$ after quantization. From the extracted regional hyper latent representation $\hat{\boldsymbol{z}}_r$, the regional synthesis transform $r_s(\cdot)$ generates the regional features $\boldsymbol{\phi}_r \in \mathbb{R}^{H \times W \times 2C}$ for contextualization.

To do this, we stack multiple layers with the downsampling and upsampling operations for the regional hyper analysis and synthesis transforms, respectively. We adopt the same structure as the previous work [22], which is based on SwinT. Specifically, the regional hyper analysis transform $r_a(\cdot)$ conducts a Patch Merge block, five SwinT blocks, a Patch Merge block, and a SwinT block. The regional hyper synthesis transform $r_s(\cdot)$ is constructed in the opposite order of the hyper analysis transform $r_a(\cdot)$, using the Patch Split block instead of the Patch Merge block.

Global context. Lastly, to capture remaining dependencies between elements across the whole image area, we construct global hyper analysis and synthesis transforms $g_a(\cdot)$ and $g_s(\cdot)$ by adopting model structure of the global hyperprior model of Informer [11]. The global hyper analysis transform $g_a(\cdot)$ extracts globally abstracted information from the latent representation using a Transformer block with cross-attention and a 1×1 convolutional layer. After quantization, it obtains a global hyper latent representation $\hat{\boldsymbol{z}}_g$. Using a 1×1 convolutional layer, the global synthesis transform $g_s(\cdot)$ infers the global features $\boldsymbol{\phi}_g \in \mathbb{R}^{N \times 2C}$ for contextualization.

3.2 Contextualize

Diverse contexts, i.e., previously modeled elements (i.e., $\hat{\boldsymbol{y}}^{<i}$) and hyper latent representations (i.e., $\hat{\boldsymbol{z}}_l$, $\hat{\boldsymbol{z}}_r$, and $\hat{\boldsymbol{z}}_g$) can be used for adapting probability distributions. Here, an important research question

https://doi.org/10.52202/079017-1461

emerges: How can we effectively leverage the diverse contexts? First, to consider step-wise varied situations, e.g., increased previously encoded/decoded elements over modeling steps, we propose a step-adaptive utilization of the three hyper latent representations. In other words, instead of applying a combined set of the three hyper latent representations, each hyper latent representation is adaptively leveraged at each step. In addition, we use regional, global, and local information sequentially. We argue that modeling order is crucial for forward adaptation, which is already known to be a key factor for backward adaptation.

The green part of Figure 2 illustrates our proposed contextualization model $c(\cdot)$ at the i-th step. First, the previously modeled $\hat{\boldsymbol{y}}^{<i}$ and the regional feature $\boldsymbol{\phi}_r$ are combined based on the same structure as in the previous approach [14], consisting of concatenation, a 1×1 convolutional layer, and three DepthConv Blocks. The DepthConv Block employs depth-wise separable convolutional layers for more efficient implementation. Second, the global feature $\boldsymbol{\phi}_g$ is combined with the output of the last DepthConv Block using the Transformer block with cross-attention [11]. Finally, we combine the output of the Transformer block with the local feature $\boldsymbol{\phi}_l$ using concatenation followed by three 1×1 convolutional layers, yielding the distribution parameters $\boldsymbol{\mu}^i$ and $\boldsymbol{\sigma}^i$. To make our contextualization model $c(\cdot)$ more efficient in terms of the number of parameters, all layers share weights across the four steps except for the initial 1×1 convolutional layer.

Discussion on modeling order. According to the study on theoretical understanding of masked autoencoder via hierarchical latent variable models, the semantic level of the learned representation varies with the masking ratio [12]. Specifically, extremely large or small masking ratios lead to low-level detailed information such as texture, while non-extreme masking ratios result in high-level semantic information. Inspired by this, we can infer that the local and global hyper latent representations correspond to relatively low-level information because they are extracted via limited utilization of the latent representation. The receptive field of the local hyper latent representation is limited by 1×1 convolutional layers. While the receptive field of the global hyper latent representation is whole image area, its attention mechanism selectively use the latent representation. Through the same reasoning, we can infer that regional hyper latent representation corresponds to relatively high-level information. Since different type of contexts has different characteristics, we argue that the modeling order is important for effective entropy modeling. In Figure 11, we empirically confirm that modeling higher-level information (i.e., regional context) first and lower-level information (i.e., local and global contexts) later is more effective than the opposite. In addition, the order between global and local is shown to be not influential.

3.3 Adapt

We design an adaptive entropy model on the quantized latent representation $\hat{\boldsymbol{y}}$ where each element is assumed to follow the Gaussian distribution, and each distribution parameters are obtained from the previous diversification and contextualization stages. Following the previous works [3, 18], we formulate our entropy model as follows:

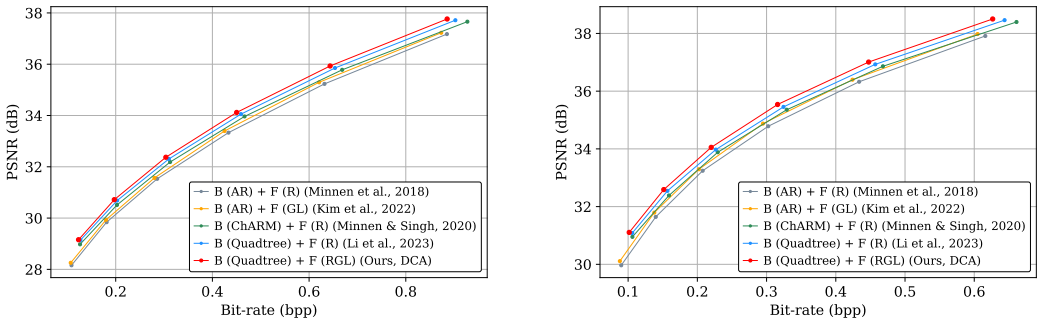
$$p_{\hat{\boldsymbol{y}}}(\hat{\boldsymbol{y}}) = \prod_i \Big(\mathcal{N}\big(\mu_i, \sigma_i^2\big) * \mathcal{U}\big(-\tfrac{1}{2}, \tfrac{1}{2}\big) \Big)(\hat{y}_i), \tag{1}$$

where μ_i and σ_i are the mean and scale of the Gaussian distribution for each element $\hat{y}_i$, respectively.

The transforms and entropy model are jointly trained in an end-to-end manner by minimizing the expected length of the bitstream (rate) and the expected distortion between the original image and the decoded image, $d(\cdot, \cdot)$. When a learned entropy model precisely matches the actual probability distribution, the entropy coding algorithm achieves the minimum rate. Therefore, we minimize the cross-entropy between the two distributions. We use mean squared error (MSE) for measuring image distortion. The objective function for our method is as follows:

$$\mathcal{L} = \mathbb{E}_{\boldsymbol{x} \sim p_{\boldsymbol{x}}} \big[-\log_2 p_{\hat{\boldsymbol{y}}}(\hat{\boldsymbol{y}}) - \log_2 p_{\hat{\boldsymbol{z}}_l}(\hat{\boldsymbol{z}}_l) - \log_2 p_{\hat{\boldsymbol{z}}_r}(\hat{\boldsymbol{z}}_r) - \log_2 p_{\hat{\boldsymbol{z}}_g}(\hat{\boldsymbol{z}}_g) + \lambda \cdot d(\boldsymbol{x}, \hat{\boldsymbol{x}}) \big], \tag{2}$$

where $p_{\boldsymbol{x}}$ is the distribution of the training dataset, the entropy models $p_{\hat{\boldsymbol{z}}_l}$, $p_{\hat{\boldsymbol{z}}_r}$, and $p_{\hat{\boldsymbol{z}}_g}$ are the non-parametric fully factorized entropy models [2], and λ is the Lagrange multiplier that determines weighting between rate and distortion. As the value increases, a model is trained in a direction that reduces information loss, and consequently leads to a higher bit-rate.



(a) (b)

Figure 4: Performance comparison with latest entropy models on the two benchmark datasets: (a) Kodak and (b) Tecnick. For clear comparisons, we denote each method as follows. B and F mean backward and forward adaptation, respectively, and the corresponding methods are written in parentheses. For backward adaptation, AR, ChARM, and Quadtree represent spatial autoregressive model, channel-wise autoregressive model, and qaudtree partition-based model, respectively. For forward adaptation, L, R, G mean local, regional, and global hyper latent representations, respectively.

4 Experiments

We use a PyTorch [19] based open-source library and evaluation platform, CompressAI [4], which has been widely used for developing and evaluating neural image codecs.

Training. We set our model parameters as follows: $C = 320$, $C_l = 10$, $C_r = 192$, and $N = 8$. We train our models corresponding six different bit-rates. We use 300,000 images randomly sampled from the OpenImages [13] dataset. We construct a batch size of 16 with 256×256 patches randomly cropped from different training images. All models are trained for 100 epochs using the Adam optimizer. The learning rate is set to 10^{-4} up to 90 epoch, and then decreases to 10^{-5}. We use PyTorch v1.9.0, CUDA v11.1, CuDNN v8.0.5, and all experiments are conducted using a single NVIDIA A100 GPU.

Evaluation. We evaluate our method on the two popular datasets: Kodak [7] and Tecnick [1]. The Kodak dataset consists of 24 images with a resolution of either 768×512 or 512×768 pixels. The Tecnick dataset is composed of 100 images with a resolution of 1200×1200 pixels. We evaluate our method in terms of rate–distortion performance. For this, we calculate the bits per pixel (bpp) after the encoding phase, and measure distortion between the decoded image and the original image using the peak signal-to-noise ratio (PSNR).

4.1 Comparison with state-of-the-art methods

We compare the proposed entropy model, DCA, with state-of-the-art entropy models. DCA can be combined with any transforms; in this paper, DCA is implemented with transforms that have the same structure as in ELIC-sm [9]. For the comparison, we further train image compression methods with four different entropy models [11, 14, 17, 18]. For a fair comparison, they are also implemented with transforms that have the same structure as in ELIC-sm [9].

Rate–Distortion. Figures 4a and 4b show the rate–distortion performance on the Kodak and Tecnick datasets, respectively. The proposed DCA consistently achieves the best rate–distortion performance across all bit-rate regions and two benchmark datasets. Specifically, DCA achieves 11.96% average rate savings over VTM-12.1 on the Kodak dataset, while the second (Lie et al. 2023) [14] and third best (Minnen & Singh, 2020) [17] methods obtain 8.55% and 4.86%, respectively.

Complexity. We also evaluate DCA in terms of efficiency. To this end, we provide the decoding time, the number of model parameters, and Bjøntegaard delta rate (BD-rate) [5] in Figure 5. Decoding time is measured on the Kodak dataset using a single NVIDIA V100 GPU.

https://doi.org/10.52202/079017-1461

Table 1: Performance comparison with state-of-the-art entropy models on Kodak.

Methods	BD-rate (%) ↓	Decoding time (ms) ↓	# Parameters (M) ↓
Baseline (CVPR'23) [14]	0.00	67.05	32.64
LIC-TCM (CVPR'23) [15]	−0.72	139.04	55.19
MLIC++ (ICMLW'23) [10]	5.76	242.61	107.80
DCA (Ours)	−3.73	82.05	37.89

BD-rate means the average bit-rate savings compared to a baseline while maintaining the same quality of decoded images. We set VTM-12.1 as the baseline, calculate BD-rate for each image in the Kodak dataset, and average them. As shown in Figure 5, our DCA achieves better rate–distortion–computation trade-off than the AR models [11, 18] and the ChARM model [17]. Even compared to the quadtree-based entropy model [14], DCA improves performance significantly, i.e., 3.73% BD-rate gain, without compromising efficiency as much as possible.

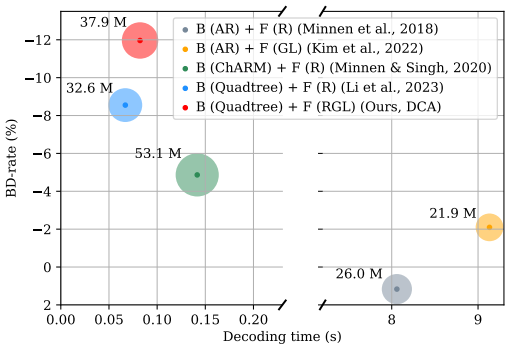


Figure 5: Performance comparison with latest entropy models on the Kodak dataset in terms of decoding time, BD-rate, and model size. Decoding time is measured on a NVIDIA V100 GPU. BD-rate means average rate savings over VTM-12.1. The size of the circle is determined proportionally to the number of model parameters, and the specific numbers are written to the left of the circles.

Using the same structure of transforms, we additionally compare DCA with two state-of-the-art entropy models (Table 1), which shows that DCA improves performance most efficiently. It is worth noting that performance improvements are significantly difficult to achieve when there are constraints on compute and memory usage, and this is the achievement of our DCA.

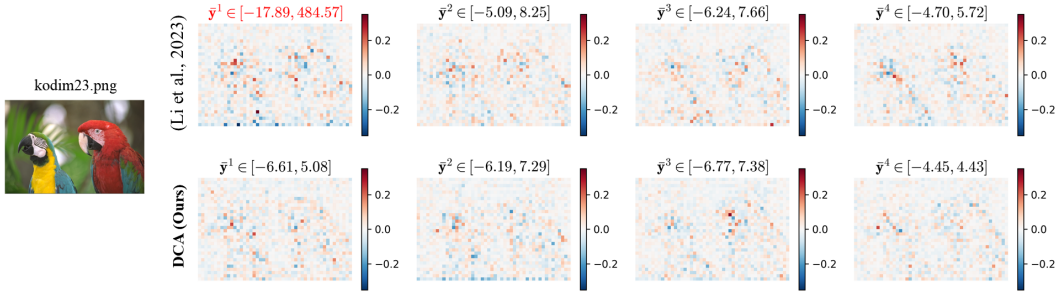


Figure 6: Illustration of normalized latent representations $\bar{\boldsymbol{y}}^i$ across four steps using both the baseline and proposed DCA. Each sub-figure includes the minimum and maximum values of normalized latent representations. Notably, the baseline exhibits a broader range of values at the first modeling step, resulting in a higher bit-rate. More examples are provided in the appendix.

Probability modeling. To further show the role of the proposed DCA, we measure the normalized latent representation for each step, i.e., $\bar{\boldsymbol{y}}^i = (\boldsymbol{y}^i - \boldsymbol{\mu}^i)/\boldsymbol{\sigma}^i$. It provides a standardized measure of how far the latent representation deviates from the predicted mean in terms of estimated standard deviations. Smaller values indicate that a learned entropy model estimates the true probability distribution more accurately. In Figure 6, we compare the result with that of the baseline model [14], which does not leverage diverse contexts for forward adaptation. Here, we have an interesting observation: While the existing work shows significantly high values at the first modeling step, DCA demonstrates consistent modeling performance across four steps. At the first step, since only forward adaptation is possible, the previous approach can utilize only a limited amount of context. On the other hand, DCA enables sufficient contextualization through diverse hyper latent representations, addressing the limitation of previous approach, which has difficulty effectively adapting to various situations.

https://doi.org/10.52202/079017-1461

Figure 7: Qualitative comparison of the decoded images by the proposed DCA and VTM-12.1.

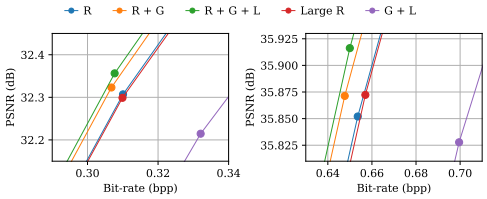


Figure 8: Analysis of forward context diversity. R, G, L denote the regional, global, and local forward contexts, respectively. "Large R" means using larger amount of the regional context.

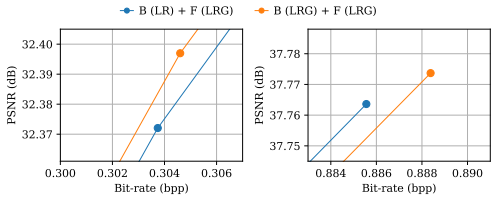


Figure 9: Analysis of backward context diversity. B and F mean backward and forward contexts, respectively. L, R, G denote the local, regional, and global contexts, respectively.

Qualitative results. We provide visual results in Figure 7, showing the decoded image from DCA has better visual quality and a higher PSNR value than that of VTM-12.1, under the same bit-rate.

4.2 Model analysis

We conduct detailed analyses of DCA. To do this, we train different variants of DCA depending on various aspects for analysis. All results are shown in two different bit-rate regions (Figures 8 to 12).

Analysis of diversification. To validate the effectiveness of diversifying contexts for forward adaptation, we compare three different methods depending on the context diversity (Figure 8): one using regional context ("R"), one using both regional and global contexts ("R + G"), and one using regional, global, and local contexts altogether ("R + G + L"). We use two additional models: one using a larger regional context ("Large R") and the other using both global and local contexts ("G + L"). The comparison among the first three methods show diversifying forward contexts is effective in both bit-rate regions. Through the results showing that the "Large R" method does not contribute to performance improvement, we once again demonstrate the effectiveness of our diversification. The last one is decomposing regional context into global and local ones rather than diversifying, which is equal to simply adopting the forward adaptation of Informer [11]. The result shows the decomposing approach significantly decreases compression efficiency, and diversifying is more effective.

In addition, while our focus lies on diversifying forward context, someone might be curious about whether the context utilized for the quadtree-based backward adaptation is diverse enough. To verify this, we conduct experiments additionally extracting global information from the backward context. Figure 9 demonstrates that there is no distinguishable advantage between two: one simply using the quadtree-based method [14] for backward adaptation ("B (LR) + F (LRG)") and the other using additional global information for backward adaptation ("B (LRG) + F (LRG)"). This implies that backward adaptation is favorable when focusing on local and regional contexts.

Analysis of contextualization. To show the effectiveness of our contextualization approach, we first compare two different methods in Figure 10. "R + G (Step-independent)" combines regional and global information in advance and utilizes the combined one regardless of the step. The other adaptively utilizes regional and global information separately for each step, "R + G (Step-adaptive)".

https://doi.org/10.52202/079017-1461

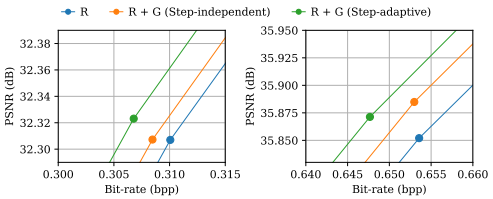


Figure 10: Analysis of how to contextualize. R and G denote the regional and global contexts, respectively. "Step-independent" first combines R and G and then utilizes them for all steps, while "Step-adaptive" does not pre-combine them and utilizes each separately for all steps.

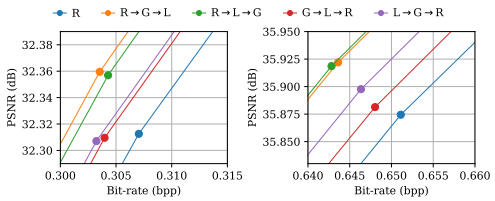


Figure 11: Analysis of contextualization order. R, G, L denote the regional, global, and local forward contexts, respectively. → means the order. For example. the "R→G→L" method utilizes R, G, and L sequentially.

As a reference, we use the model utilizing only regional context, i.e., "R". The result shows that both are effective and our step-adaptive approach is more beneficial for boosting performance.

In addition, we analyze the effectiveness of our modeling order for contextualization in Figure 11. Four different ordering methods and one reference method are used for the comparison. Our ordering approach ("R→G→L") achieves the best rate–distortion performance, and the methods are categorized into two groups based on the performance in both bit-rate regions. Models that prioritize the regional context ("R→G→L" and "R→L→G") show better performance compared to those that do not prioritize it ("G→L→R" and "L→G→R"). We observe that the method using the opposite modeling order of the proposed sequence ("L→G→R") even exhibits a performance decline compared to the baseline ("R") in a lower bit-rate region.

Analysis of architecture. Applying attention mechanisms on various tasks is one of the most actively researched topics. We analyze the effect of the architecture combination for forward and backward adaptation in DCA. Figure 12 compares three different methods: one without attention ("B (CNN) + F (CNN)"), another applying attention only to forward adaptation ("B (CNN) + F (Attnention)"), and the last applying attention for both ("B (Attention) + F (Attention)"). The results show that CNN and attention are effective for forward and backward adaptation, respectively. We infer that focusing on local and regional information is preferable in backward adaptation; thus, a CNN with a locality inductive bias may be more effective.

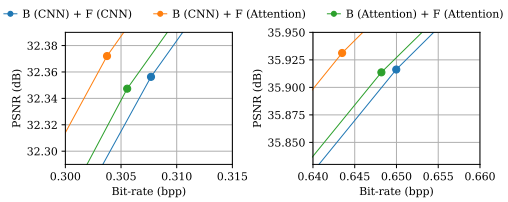


Figure 12: Analysis of model architecture for backward and forward adaptation. B and F mean backward and forward adaptation, respectively.

5 Conclusion

In this paper, we proposed a fast and effective entropy modeling framework, DCA, which diversifies forward contexts by extracting local, regional, and global information, and contextualizes current elements with the diverse forward and backward contexts. We demonstrated that our DCA improves rate–distortion performance significantly compared to previous approach without compromising efficiency as much as possible. Furthermore, we provided diverse insights into entropy modeling by conducting a comprehensive and in-depth analysis of the design aspects of DCA.

Limitation and future works. To address the limitation of the state-of-the-art entropy models, we focused on paving a novel framework with diverse contexts rather than designing neural architectures. Therefore, DCA can be limited by the architectural designs that are inspired by the existing works [11, 14, 22]. In the future, we expect that it would be further improved by neural architectures especially designed for the diverse contexts. In addition, it is worth exploring alternative criteria for diversification beyond the spatial range the contexts covers (i.e., local, regional, and global contexts).

https://doi.org/10.52202/079017-1461

References

[1] N. Asuni and A. Giachetti. TESTIMAGES: a large-scale archive for testing visual devices and basic image processing algorithms. In *Proceedings of the Smart Tools and Apps for Graphics (STAG)*, 2014.

[2] J. Ballé, V. Laparra, and E. P. Simoncelli. End-to-end optimized image compression. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2017.

[3] J. Ballé, D. Minnen, S. Singh, S. J. Hwang, and N. Johnston. Variational image compression with a scale hyperprior. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018.

[4] J. Bégaint, F. Racapé, S. Feltman, and A. Pushparaja. CompressAI: a PyTorch library and evaluation platform for end-to-end compression research. *arXiv preprint arXiv:2011.03029*, 2020.

[5] G. Bjøntegaard. Calculation of average PSNR differences between RD-curves. *VCEG-M33*, 2001.

[6] J. Duda. Asymmetric numeral systems: entropy coding combining speed of Huffman coding with compression rate of arithmetic coding. *arXiv preprint arXiv:1311.2540*, 2013.

[7] R. Franzen. Kodak lossless true color image suite. http://r0k.us/graphics/kodak/, 1999.

[8] D. He, Y. Zheng, B. Sun, Y. Wang, and H. Qin. Checkerboard context model for efficient learned image compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[9] D. He, Z. Yang, W. Peng, R. Ma, H. Qin, and Y. Wang. 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)*, 2022.

[10] W. Jiang and R. Wang. MLIC++: Linear complexity multi-reference entropy modeling for learned image compression. In *Proceedings of the International Conference on Machine Learning (ICML) Workshop*, 2023.

[11] J.-H. Kim, B. Heo, and J.-S. Lee. Joint global and local hierarchical priors for learned image compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[12] L. Kong, M. Q. Ma, G. Chen, E. P. Xing, Y. Chi, L.-P. Morency, and K. Zhang. Understanding masked autoencoders via hierarchical latent variable models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[13] I. Krasin, T. Duerig, N. Alldrin, V. Ferrari, S. Abu-El-Haija, A. Kuznetsova, H. Rom, J. Uijlings, S. Popov, A. Veit, S. Belongie, V. Gomes, A. Gupta, C. Sun, G. Chechik, D. Cai, Z. Feng, D. Narayanan, and K. Murphy. OpenImages: A public dataset for large-scale multi-label and multi-class image classification. *Dataset available from https://github.com/openimages*, 2017.

[14] J. Li, B. Li, and Y. Lu. Neural video compression with diverse contexts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[15] J. Liu, H. Sun, and J. Katto. Learned image compression with mixed Transformer-CNN architectures. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[16] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo. Swin Transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[17] D. Minnen and S. Singh. Channel-wise autoregressive entropy models for learned image compression. In *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, 2020.

[18] D. Minnen, J. Ballé, and G. Toderici. Joint autoregressive and hierarchical priors for learned image compression. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[19] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[20] Y. Qian, Z. Tan, X. Sun, M. Lin, D. Li, Z. Sun, L. Hao, and R. Jin. Learning accurate entropy model with global reference for image compression. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.

 https://doi.org/10.52202/079017-1461

[21] Y. Qian, X. Sun, M. Lin, Z. Tan, and R. Jin. Entroformer: A transformer-based entropy model for learned image compression. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2022.

[22] Y. Zhu, Y. Yang, and T. Cohen. Transformer-based transform coding. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2022.

[23] R. Zou, C. Song, and Z. Zhang. 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)*, 2022.

A Additional results

We provide the normalized latent representations of images from the Kodak dataset in Figure 13.

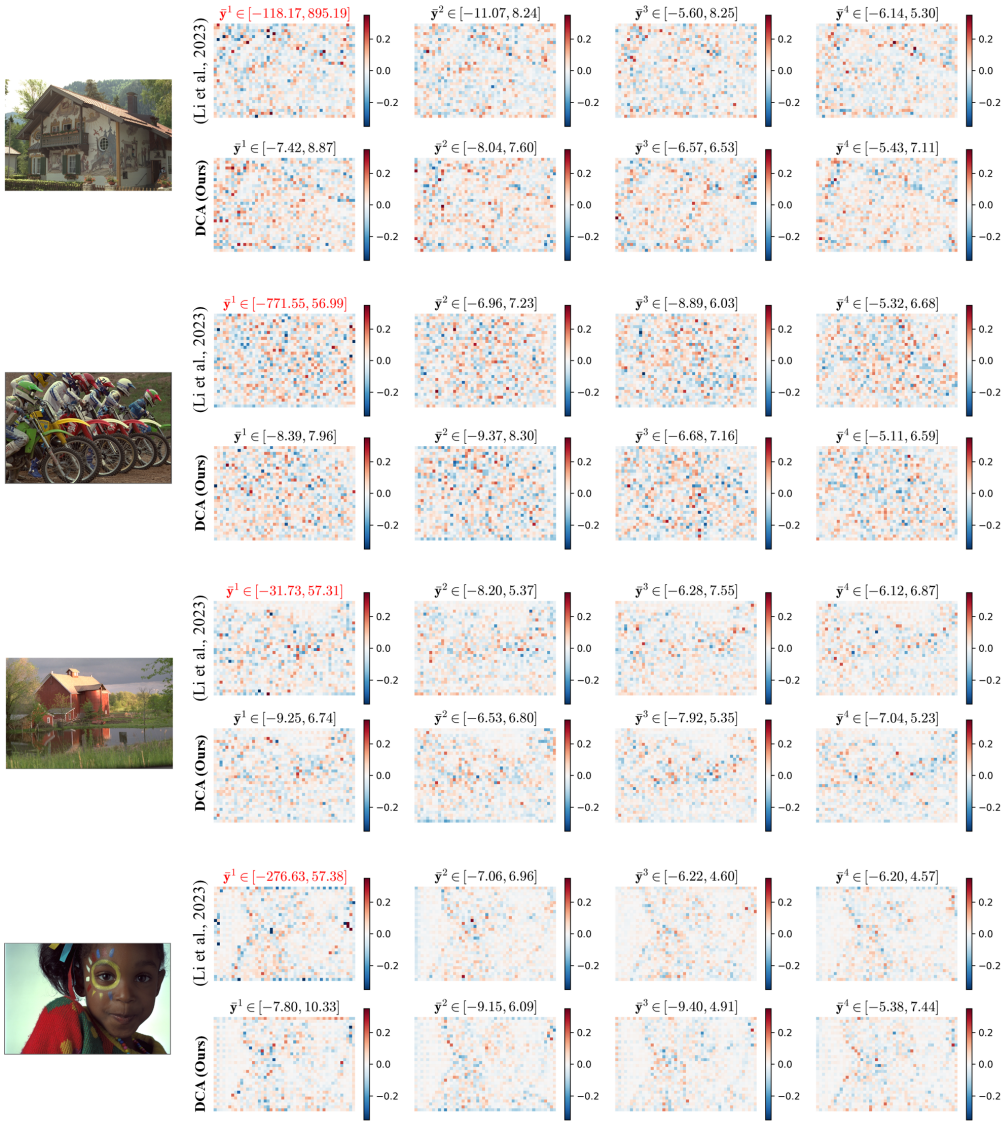


Figure 13: Illustrations of normalized latent representations $\bar{\boldsymbol{y}}^i$ across four steps for Kodak images using both the baseline [14] and proposed DCA. Each sub-figure includes the minimum and maximum values of normalized latent representations. Notably, the baseline exhibits a broader range of values at the first modeling step, resulting in a higher bit-rate.

We provide an in-depth runtime analysis by sub-systems in Table 2.

Table 2: Runtime (ms) of DCA. Total encoding/decoding time includes the ANS entropy coding.

Transforms		Entropy model (DCA)							Total	
f_a	f_s	l_a	l_s	r_a	r_s	g_a	g_s	c	Encoding	Decoding
3.46	1.54	0.98	0.96	5.32	4.78	0.63	0.08	14.75	117.49	82.05

https://doi.org/10.52202/079017-1461

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We summarized our contributions in Section 1 and illustrated our method in Figure 1 in a compact manner.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: See Section 5.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: Our method is evaluated based on empirical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: See Figure 2 and Sections 3 and 4.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

https://doi.org/10.52202/079017-1461

Answer: [No]

Justification: The code is proprietary.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: See Section 4.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We evaluate our method across six different bit-rate regions and our method consistently achieves performance improvements.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: See Section 4.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: Our work conforms the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: We think our work is foundational research.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

https://doi.org/10.52202/079017-1461

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: Our work does not pose such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: See Section 4.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: We do not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: Our paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: Our paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

https://doi.org/10.52202/079017-1461