# Cross-Granularity Online Optimization with Masked Compensated Information for Learned Image Compression

Haowei Kuang[1] Wenhan Yang[2] Zongming Guo[1] Jiaying Liu[1]✉
[1]Wangxuan Institute of Computer Technology, Peking University, Beijing, China
[2]Pengcheng Laboratory, Shenzhen, China
kuanghw@stu.pku.edu.cn, yangwh@pcl.ac.cn, {guozongming, liujiaying}@pku.edu.cn

## Abstract

*Learned image compression aims to reduce redundancy by accurately modeling the complex signal distribution inherent in images with network parameters. However, existing practices that train models on entire dataset offline face a limitation, as the estimated distribution only approximates the general image signal distribution and fails to capture image-specific characteristics. To address this issue, we propose a cross-granularity online optimization strategy to mitigate information loss from two key aspects: statistical distribution gaps and local structural gaps. This strategy introduces additional fitted bitstream to push the estimated signal distribution closer to the real one at both coarse-grained and fine-grained levels. For coarse-grained optimization, we relax the common bitrate constraints during gradient descent and reduce bitrate cost via adaptive QP (Quantization Parameter) selection, preventing information collapse and narrowing the statistical distribution gaps. For fine-grained optimization, a Mask-based Selective Compensation Module is designed to sparsely encode structural characteristics at low bitrates, enhancing local distribution alignment. By jointly optimizing global and local distributions, our method achieves closer alignment to real image statistics and significantly enhances the performance. Extensive experiments validate the superiority of our method as well as the design of our module. Our project is publicly available at: https://ellisonkuang.github.io/CGOO.github.io/.*

## 1. Introduction

Image compression aims to reduce storage space and bandwidth requirements while retaining as much critical visual information as possible. It grows increasingly significant with the ever-increasing demand for high-resolution and high-quality images. The essence of image compression is to estimate the distribution of the source signal by accurately modeling the complex real-world signal distribution unique to each image.

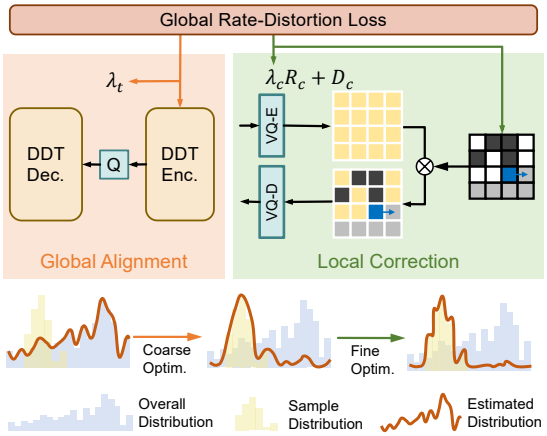


Figure 1. Workflow of our Cross-Granularity Optimization with both coarse and fine-grained optimization. The coarse-grained optimization aligns the real distribution with the estimated sample distribution of images on a global scale, while the fine-grained optimization further refines the alignment to minimize the gap by compensating for local details.

In recent decades, conventional image compression methods like JPEG [37], JPEG2000 [28], and BPG [6] have been widely used, which utilize key steps such as transformation, quantization, entropy coding and some prediction methods to estimate the real distribution and remove redundant information for efficient transmission. With the rapid development of deep learning, many learned image compression methods [11, 15, 25, 40] based on neural networks are proposed. The pioneering approaches investigate using Generalized Divisive Normalization (GDN)-embedded transform networks [2, 5] or develop recurrent architectures [35] to enable variable-rate compression. Later research focuses on integrating more advanced modules, such as attention mechanisms [11] and transformer-based architectures [45, 46], to enhance learned image compression. Additionally, efforts have been made to improve the efficiency of entropy coding, including the development of

joint autoregressive and hyperprior models [31] along with their related variants [29]. Different from conventional approaches, these learned image compression methods replace the step-by-step manual optimization with an end-to-end learned network, achieving better rate-distortion optimization performance.

However, both conventional methods based on manual design and end-to-end methods that optimize rate-distortion cost on entire dataset face challenges in perfectly matching the distribution of each input image, resulting in a tendency to fit average attributes rather than individual images, which is called amortization effect [12]. Specifically, this gap is reflected in two aspects: 1) the global gap caused by differences in content and theme of images; 2) the local gap related to fine structure and texture information, caused by the quantization and randomness of the sampling process.

For the two levels of the distribution gap, conventional coding techniques employ several technologies to transmit additional pattern information or distribution parameters, such as Adaptive Loop Filter (ALF) [44] for addressing the global gap and Multiple Transform Selection (MTS) for the local structural distribution gap, enabling sample-adaptive modeling to reduce rate-distortion cost. And for the learned image compression, some methods are proposed to perform per-sample online optimization through online gradient descent [38, 42] for each image or adaptive ensemble multiple models [39]. However, in these methods, per-sample optimization is performed only at the image level, aiming to narrow the global gap. They do not consider the fine-grained local structure distribution gap, which limits the integration and enhancement of learning capability and fine detailed signal representation.

Our work aims to fill up this blank by introducing a **Cross-Granularity Online Optimization** strategy into learned image compression. Our general idea is to incorporate additional fitted bitstreams to bring the estimated signal distribution closer to the real one at both coarse and fine-grained levels. In detail, for fine-grained optimization, we introduce a sparse representation strategy and design a Mask-based Selective Compensation Module to achieve it in networks. By encoding diverse structural distribution characteristics with network parameters, the signals can be reconstructed with a sparse representation at a lower bitrate cost, efficiently compensating for fine-grained structural distribution information. For coarse-grained optimization, we further improve the existing gradient descent based method, relax the common bitrate constraints in gradient descent. We propose to reduce bitrate cost via adaptive QP selection, preventing information collapse and realizing the optimization of image-level distribution. By leveraging a joint global-local online optimization, our method achieves more accurate alignment between the estimated and real distributions and enhances the performance on each images.

Our contributions are summarized as follows:

- We propose a Cross-Granularity Online Optimization strategy for learned image compression. The coarse-grained per-sample optimization aligns the distribution of the reconstructed signal to the original one at the image level, and the fine-grained optimization further narrows the distribution gap in a dense manner with structural detail compensation, jointly achieving a more accurate alignment between the estimated and real distributions.
- For fine-grained optimization, we design a Mask-based Selective Compensation Module, which learn to encode diverse structural distribution characteristics with neural networks' parameters. Therefore, during inference-time optimization, the signals can be reconstructed using a sparse representation, further reducing the rate-distortion cost while efficiently compensating for fine-grained structural distribution information.
- For coarse-grained gradient optimization, we propose a progressive bitrate contrainst strategy. By relaxing the constraint of bitrate cost in training and decreasing it through adaptive QP selection in inference, the risk of information collapse caused by a conventional variational rate-distortion constraint are mitigated.

## 2. Related work

### 2.1. Learned Image Compression

Due to significant advancements in deep learning, in recent years, deep learning-based image compression techniques have surpass traditional methods in achieving an optimal balance between bit rate and reconstruction quality. Initially, Ballé et al. [2, 3] were pioneers in using neural networks to build lossy image compression autoencoders, which sparked a wave of learned image compression methods. Many of these efforts expect to use the ability of neural networks to fit real image distribution for removing more information redundancy through more efficient transformations. Based on this idea, more efficient network architectures are explored, including CNN-based architectures [30, 32], Transformer-based architectures [20, 24, 26], Transformer-CNN hybrid architectures [25] and so on. Beyond transformations, many studies have concentrated on entropy coding of latent representations based on learned probability models, such as hyper-priors [4] and context models [10, 23]. Additionally, the incorporation of Gaussian Mixture Models and attention-based modules further boosted the performance of image compression [11].

In addition, there are some VQ-Based methods [16, 27, 41] that store diversified distribution through codebook based on neural network parameters, enabling the selection of suitable representations for different samples. However, due to the limitation of codebook's capacity and optimization strategy, most of these methods are only suitable

for ultra-low bitrates, lack the capacity of improving image quality with increased bitrate. Unlike the previous methods, our approach integrates end-to-end compression framework with a VQ-based module. It delivers the primary content through the end-to-end framework and sparse compensation data via the codebook, enhancing the codebook's utility across a broader range of bitrates.

### 2.2. Optimization in Image Compression

To tailor the compression method to each sample, an effective strategy is to perform per-sample online optimization on each sample that needs to be compressed. In conventional hybrid coding frameworks, an extensive online optimization is carried out for every sample. For instance, in the latest VVC coding framework [8], Multiple Transform Selection (MTS) is introduced to select the most desirable transform, and 67 modes with diverse reference pixels are utilized for intra-frame prediction, which taking into account the unique features of each sample's local position.

In terms of deep learning-based approaches, there is also many work trying to further perform per-sample optimization under well-optimized global distributions to further optimize performance. For example, [17] optimizes the encoder by gradient descent, [39] uses ensemble learning to select a transform model from a pool of models to compress the image, and [38] optimizes both encoder and decoder by transmitting additional model stream. However, most of these methods only perform optimization at the image level, and do not fully consider the local structural features inside the image. In addition, there are some efforts based on implicit neural representation [9, 22, 34] to achieve fine optimization of image details by storing images in model parameters through iterative optimization, but it is difficult to apply to large-resolution images due to its high optimization cost. In our work, we address the limitations of the the above methods by designing a cost-effective approach for cross-granularity optimization in neural networks.

## 3. Cross-Granularity Online Optimization

### 3.1. Preliminaries and Motivations

We start with the theoretical analyses of the rate distortion theory following the symbolic expression of [43]. In Shannon's seminal work [33], the rate-distortion function $R(D)$ for the given random variable source $X$ in distribution $P_X$ and distortion measure $\Delta(\cdot,\cdot)$ is defined as:

$$\begin{aligned} R(D) &= \inf_{P_{\hat{X}|X}} I(X;\hat{X}), \\ \text{subject to: } & \mathbb{E}_{P_X P_{\hat{X}|X}}[\Delta(X,\hat{X})] \le D, \end{aligned} \tag{1}$$

where $\hat{X}$ means the reconstruction characterized by distribution $P_{\hat{X}}$, $I(\cdot;\cdot)$ is the mutual information and $\mathbb{E}(\cdot)$ is the mathematical expectation. This function describes the inferior limit of bitrate under a given distortion threshold $D$.

In a practical codec, considering the definitive sample $x$ in distribution $P_X$, we can reformulate Eq. (1) through Lagrangian relaxation as an unconstrained optimization problem. Besides, the distribution $P_{\hat{X}}$ depends on the source distribution and is unknown on the decoder side. Due to the difficulty of modeling the distribution directly, most existing methods choose to introduce a new variable $Y$ and reformulate rate distortion function to:

$$\begin{aligned} L(P_{Y|X}, Q_Y, P_{\hat{X}|Y}) = \mathbb{E}_{x\sim P_X}[KL(P_{Y|X=x} \| Q_Y)] \\ + \lambda \mathbb{E}_{P_X P_{Y|X} P_{\hat{X}|Y}}[\Delta(X,\hat{X})], \end{aligned} \tag{2}$$

where $Q_Y$ denotes the modeling of distribution $P_Y$ at the decoder side, $KL(\cdot\|\cdot)$ denotes the Kullback-Leibler divergence and $\lambda$ denotes a hyper-parameter which can adjust the trade-off between rate and distortion. After that, in most existing learning-based compression methods, encoder $F_E$, decoder $F_D$ and entropy coding model are parameterized by neural networks to fit $P_{Y|X}$, $P_{\hat{X}|Y}$ and $Q_Y$, which means $Y = F_E(X|\phi_E)$ and $\hat{X} = F_D(Y|\phi_D)$, where $\phi_E$ and $\phi_D$ are network parameters. Then, the objective becomes:

$$\begin{aligned} L(F_E, Q_Y, F_D) = \mathbb{E}_{P_X}[-\log Q_Y(F_E(X|\phi_E))] \\ + \lambda \mathbb{E}_{P_X}[\Delta(X, F_D(F_E(X|\phi_E)|\phi_D)]. \end{aligned} \tag{3}$$

The network parameters are optimized with Eq. (3) and shared across the entire dataset, rather than directly optimizing $Y$ for each sample $x$.

Although theoretically feasible to use enough parameters in $\phi_E$ and $\phi_D$ to fully fit the distribution $P_{Y|X}$ and $P_{\hat{X}|Y}$, in practice it is difficult to achieve due to the expensive costs and limitations of the neural network capabilities. In fact, the estimated distribution typically only roughly fits the general image signal distribution, falling short of perfectly matching the real distribution of each sample, which causes the amortization effect [12].

To alleviate this problem, many strategies are proposed to perform per-sample online optimization on $F_E$ for each sample based on well-optimized $\phi_E$. Furthermore, an effective strategy [38] introduces an additional latent variable $S$, through which $F_D$ can also be optimized based on $\phi_D$. The per-sample optimization of $F_D$ can be performed by gradient descent with the objective:

$$s = \arg\min_s -\log Q_S(s) + \lambda \Delta\left(x, F_D(y|\phi_D, s)\right), \tag{4}$$

where $s$ and $y$ are specific samples in random variables $S$ and $Y$ respectively, and $Q_S$ is the estimated distribution of S. However, there are two notable issues:

- **Absence of optimization for local structural distribution.** Per-sample optimization through gradient descent

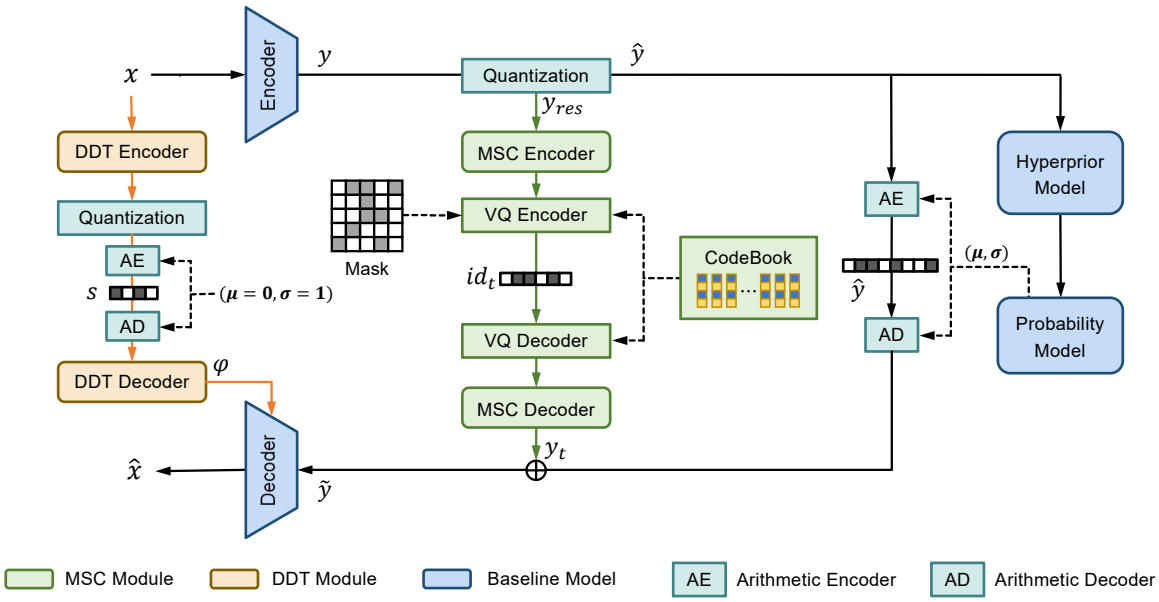


Figure 2. The overall structure of our proposed image compression framework with cross-granularity optimization.

is expensive, so it can only be optimized at the entire image level at most. However, fitting only at the image level leaves gaps in the local structural distribution, where fine structure loss and sampling randomness hinder optimization through gradient descent alone.

- **Model collapse caused by vanilla variational objective.** In practice optimization, due to the different dynamics of $y$, $s$, the existing gradient descent strategy with objective Eq. (4) might lead to the model collapse of $s$, making it difficult to learn useful information in $s$, and the optimization results are unable to achieve the optimal rate-distortion tradeoff.

In our work, we aim to adopt a novel method to address both of these issues:

- For the first issue, we propose to combine an additional fine-grained online optimization with the gradient descent based coarse-grained optimization. On the basis of coarse-grained gradient optimization which aligns the distribution at the image level, the distribution gap is further reduced effectively in a dense way by structural detail compensation in fine-grained online optimization.
- For the second issue, we propose to relax the bitrate cost constraint of $s$ during gradient descent and decrease its cost through adaptive quantization parameter search, achieving a better rate-distortion trade-off.

The following sections describe our method in detail.

## 3.2. Cross-granularity Compensation Framework

### 3.2.1. Overall Structure

Our work builds on existing end-to-end image compression framework [25] with hyper-prior for entropy estimation. Beyond the baseline, we introduce a Mask-based Selective Compensation (MSC) module and a Data-Dependent Transform (DDT) module, through which we perform coarse-grained and fine-grained online optimization, respectively. The whole structure is shown in Fig. 2, in which the MSC and DDT modules are marked as green and yellow.

For an image $x$ to be encoded, we first perform inference with the end-to-end encoder $\mathrm{E}(\cdot)$ and quantize it, resulting in an quantized latent $\hat{y}$:

$$\hat{y} = \mathrm{Q}(\mathrm{E}(x)), \tag{5}$$

where $\mathrm{Q}(\cdot)$ means quantizer.

Next, we extract and transmit a syntax vector from the original image $x$ to generate adaptive parameters $\varphi$ at the decoder side using the Data-Dependent Transform (DDT) Module, which is described below. In this module, gradient descent based coarse-grained online optimization will be performed on extracted syntax vector $s$ to make it learn the characteristics of the image-level distribution of sample $x$. Then, parameters $\varphi$ will be applied to the decoder to realize the approximation between the estimated and the real distribution at the image-level through sample dependence transformation:

$$\varphi = \mathrm{DDT}(x). \tag{6}$$

After that, we calculate the residual before and after quantization $y_{res}$, and obtain the compensated feature $y_t$ through the Mask-based Selective Compensation (MSC) Module. This compensation feature is transmitted through the index map $id_t$. In this module, fine-grained optimization is performed by mask-based selection to obtain the most ef-

fective compensation $y_t$ of fine-grained structure distribution information, and it is combined with quantified latent $\hat{y}$ to compensate for local details and realize the approximation of the estimated and the real distribution at the local structure level:

$$\begin{aligned} y_{res} &= y - \hat{y}, \\ y_t &= \mathrm{MSC}(y_{\mathrm{res}}). \end{aligned} \tag{7}$$

Finally, the compensated feature $y_t$ and $\hat{y}$ are combined at the decoding side to get $\tilde{y}$ and reconstruct image $\hat{x}$ with end-to-end decoder $\mathrm{D}(\cdot)$ with the help of parameters $\varphi$ extracted by DDT module:

$$\begin{aligned} \tilde{y} &= \hat{y} + y_t, \\ \hat{x} &= \mathrm{D}(\tilde{y}|\varphi). \end{aligned} \tag{8}$$

In summary, our method considers three bitstreams that need to be transmitted: quantized latent $\hat{y}$, index map $id_t$ and syntax vector $s$. Our method uses an existing hyperprior model, so we do not go into its details here. Our overall optimization goal is:

$$L = R(\hat{y}) + R(s) + R(id_t) + \lambda\Delta(x, \hat{x}), \tag{9}$$

where $R(\cdot)$ denotes the bit rate cost and we use mean-squared-error as distortion measure in our method, which means $\Delta(x, \hat{x}) = \|x - \hat{x}\|^2$. These definitions will be used later in the introduction.

#### 3.2.2. Data-Dependent Transform Module

The network structure of DDT module is similar to [38] and [21]. Specifically, in DDT encoder $\mathrm{E}_{\mathrm{DDT}}$, features at multiple scales are extracted by multi-layer convolution, which are globally average pooled and concatenated into a compact one-dimensional vector $s$. In this way, the extracted features have multi-scale information and globally consistent characteristics at the same time, which is more suitable for fitting the distribution at the image level. Then, unlike [38], the vector $s$ is quantized through a quantizer $Q(\cdot, \cdot)$ with adjustable quantization parameter $\lambda_t$, and performs compression with an arithmetic codec based on standard Gaussian distribution. Finally, $s$ generates a set of adaptive convolution kernel parameters $\varphi$ through the DDT decoder $\mathrm{D}_{\mathrm{DDT}}$, which is a multi-layer perceptron with residual. And $\varphi$ acts as parameters of the last convolution layer of decoder D. The full process is expressed as:

$$\begin{aligned} s &= Q(\mathrm{E}_{\mathrm{DDT}}(x), \lambda_{\mathrm{t}}), \\ \varphi &= \mathrm{D}_{\mathrm{DDT}}(s). \end{aligned} \tag{10}$$

#### 3.2.3. Mask-based Selective Compensation Module

Mask-based Selective Compensation module contains a pair of convolution-based autoencoders $\mathrm{E}_{\mathrm{MSC}}(\cdot)$ and $\mathrm{D}_{\mathrm{MSC}}(\cdot)$, a pair of vector quantization codec modules $\mathrm{E}_{\mathrm{VQ}}$ and $\mathrm{D}_{\mathrm{VQ}}$, and a codebook $Z = \{z_k\}_{k=1}^{K} \in \mathbb{R}^n$ that stores diverse structural distribution characteristics.

In the MSC module, we first perform pre-selection. After obtaining the compensation feature $y_c \in \mathbb{R}^{h \times w \times n}$ by the MSC Encoder $\mathrm{E}_{\mathrm{MSC}}$, each pixel in the feature will find the corresponding quantization vector in the codebook via the $\mathrm{E}_{\mathrm{VQ}}$ and get the quantized compensation feature $\hat{y}_c$ and the corresponding index map $id_c$:

$$\hat{y}_c, id_c = \underset{z_k \in Z, id_k}{\arg\min} \|z_k - y_c^{ij}\| + \lambda_c R(id_k), \tag{11}$$

where $\lambda_c$ denotes the trade-off parameters which will be determined by online optimization.

We then perform a sparse masking operation on the pre-selection results to selectively transmit the most beneficial information, thereby achieving greater performance gains. A sparse representation mask $M$ is then generated by fine-grained optimization and computes the sparse compensation representation $id_t$:

$$id_t = id_c \odot M, \tag{12}$$

where $\odot$ means element-wise multiple. Considering the sparsity of $id_t$, it is encoded by global probability distribution $Q_t$ when it is transmitted. Finally, $\mathrm{D}_{\mathrm{VQ}}$ looks up the codebook $Z$ to get the corresponding vector through the $id_t$ to form the sparse compensation feature $\tilde{y}_c$.

### 3.3. Cross-Granularity Optimization Strategy

As mentioned above, our method employs both coarse-grained and fine-grained online optimization strategies during inference, aiming to minimize the gap between the predicted distribution and the real distribution under two levels of signal distribution, *i.e.*, image-level and local structure distributions, respectively.

#### 3.3.1. Coarse Optimization

In coarse-grained optimization, our target is to minimize the gap between the estimated general distribution and the real distribution of a single sample at the image level. Similar to most existing methods, we performed coarse online optimization on network parameters through a gradient descent strategy to optimize network parameters at the image-level. In order to avoid the model collapse mentioned above, we split the strategy of performing gradient descent with Eq. (4) as the optimization objective function in the previous method into two stages.

Firstly, in the process of gradient optimization on DDT Encoder, the bitrate cost constraint on syntax $s$ is relaxed, and the optimization objective is modified from Eq. (4) to:

$$s = \underset{s}{\arg\min}\, \Delta(x, F_D(y|\phi_D, \mathrm{D}_{\mathrm{DDT}}(s))). \tag{13}$$

Then, starting with a large initial value, a binary search algorithm is employed to iteratively adjust the quantization

parameter $\lambda_t$, aiming to achieve the optimal trade-off between the syntax vector's bitcost and performance.

#### 3.3.2. Fine Optimization

In fine-grained optimization, we focus on the local structural details of the image. Due to the quantization and inherent randomness involved in the image sampling process, the image exhibits discrete characteristics in these local detail distributions. This property makes it impossible for gradient descent methods to further fit these local structural details on the training data in an offline setting. In order to achieve online optimization on local structural details, we propose to use a sparse representation strategy to compensate for these structural details. The optimization process is divided into the following steps:

**Pre-Selection Tradeoff Optimization.** In the process of sparse compensation with the MSC module, we need to balance the cost of sending compensation features with the benefits they bring. In the pre-selection stage, since the feature $y_c^{ij}$ is the vector that can achieve the maximum performance gain without considering the transmission cost, for a certain feature $y_c^{ij}$, the closer the matching vector $z_k$ from the codebook is to $y_c^{ij}$, the more benefit it might given, so the benefits can be represented by the L1 norm of $y_c^{ij}$ and $z_k$. However, this L1 norm serves only as an approximate measure of performance gains and does not establish a strict positive correlation with $\Delta(x, \tilde{x})$. Thus, when balancing the rate-gain trade-off, it is not possible to directly ascertain a suitable $\lambda_c$ that aligns perfectly with $\lambda$ in Eq. (3). In order to achieve the alignment of $\lambda_c$ and $\lambda$, we perform a binary search strategy to find $\lambda_c$ with the lowest global Rate-Distortion Loss.

**Sparse Masking Optimization.** Through the pre-selection process, we search for a corresponding quantized vector for the features of each position. However, not all vectors can provide sufficient positive effects for correcting local structural information compensation, and the gains brought by a large part of the vectors cannot cover the losses caused by their rate costs. To achieve optimal cost performance in transmitting compensation information, we design a sparse mask optimization strategy that masks vectors offering insufficient gain.

The sparse masking optimization operates for each pixel in the compensation feature map $\hat{y}_c$. First, the compensation feature is independently injected into position a through element-wise addition. Then, the decoder D reconstructs the image with this modified feature, enabling calculation of the loss differential between pre- and post-compensation states. This performance then determines the binary masking decision: contributing positions (mask value = 1) activate feature transmission, while non-contributing positions (mask value = 0) are pruned from the transmission to optimize bitrate cost.

### 3.4. Progressive Multi-Stage Training Strategy

To enhance the alignment between end-to-end neural network parameters and real-world data distributions while establishing an expanded optimization potential for cross-granularity online optimization strategies, we propose a progressive training strategy consisting of sequentially optimized multiple training stages. We start the training by loading the pre-trained model parameters of the baseline end-to-end network, then we train the DDT module on the basis of these parameters, and ultimately conclude with training MSC module.

**Training Strategy of DDT Module.** Similar to coarse optimization, we also relax the rate constraint on syntax vector $s$ during gradient descent based training process. The loss function is as follows:

$$L_{DDT} = -\log Q_Y(y) + \lambda\Delta(\tilde{x}, F_D(F_E(x)|\phi_D, s)). \quad (14)$$

**Training Strategy of MSC Module.** The training process of the MSC module comprises two parts: the training of the MSC Encoder/Decoder and the training of the codebook.

Initially, we focus on training the MSC Encoder and Decoder. To do this, we remove the VQ Encoder $\mathrm{E_{VQ}}$ and the VQ Decoder $\mathrm{D_{VQ}}$ from the network architecture, which eliminates the vector quantization operation applied to $y_c$. With these components removed, we keep the rest of the network fixed and train the MSC Encoder/Decoder independently. The loss function utilized for this training is the Mean Squared Error (MSE) between the reconstructed image and the input image. After completing this training phase, the MSC Encoder/Decoder can extract the relevant compensation information from the residual $y_{res}$.

Next, we proceed with training the codebook $Z$. Our approach adopts the training strategy in the existing work [13], wherein the gradient is propagated from the decoder to the encoder [7], enabling end-to-end training of both the model and the codebook through the loss function:

$$\begin{aligned} L_{MSC} = \|x - \hat{x}\|^2 + \|sg[E_{MSC}(y_{res})] - z_q\|_2^2 \\ + \|sg[z_q] - E_{MSC}(y_{res})\|_2^2, \end{aligned} \quad (15)$$

where $z_q$ represents the vector selected from the codebook and $sg[\cdot]$ denotes the stop-gradient operation. To maximize the activation of vectors in the codebook and learn a diverse and rich structural distribution, we implement a warm-up training phase prior to selecting the vector $z_q$ using the strategy outlined in Eq. (13). During this warm-up period, $z_q$ is randomly chosen from the codebook to minimize the distance between all the vectors in the codebook and $y_c$, thus mitigating the issue of low codebook activation rate.

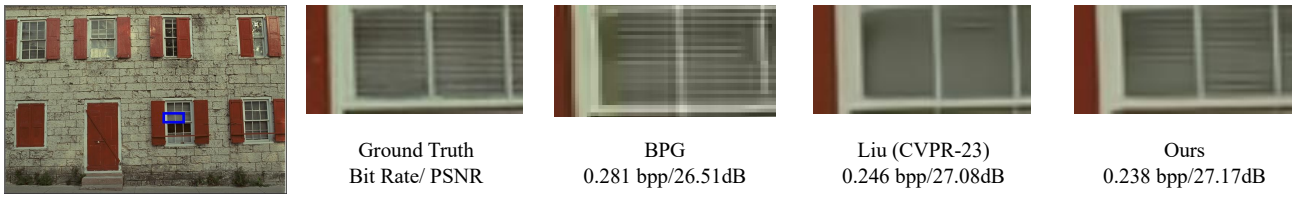


Figure 3. Visual results compared to BPG [6] and Liu (CVPR-23) [25].

# 4. Experiments

## 4.1. Implementation

**Network Implementation.** Specifically, we implement our network on the basis of existing end-to-end learning based image compression methods [25], and the parameter setting follows its small model setting (channel number of TCM block is 128) which is completely open source. The codebook size $K$ is set to 4096, which is an appropriate value based on experimental attempts. In order to reduce the correlation between pixels of sparse representation $id_t$ and avoid the effect of partially masked pixels on the reconstruction of surrounding pixels, MSC Encoder and decoder use multiple layers of $1 \times 1$ convolution without bias. The detailed structure and hyper-parameters of the networks are shown in the supplementary material.

**Training Details.** We use DIV2K image dataset [1] as our training dataset, which contains 800 high-quality natural images with an average 2K resolution. We use the Adam optimizer [18] in each phase of the training, and train 5 models with different compression rates based on different bit rates end-to-end baseline parameters [25], with $\lambda$ in $\{2.5 \times 1^{-3}, 3.5 \times 1^{-3}, 6.7 \times 1^{-3}, 1.3 \times 1^{-2}, 2.5 \times 1^{-2}\}$. For the global probability distribution $Q_t$, we randomly select 50 images in the training dataset as a validation set, and calculate the distribution of $id_c$ as an estimate of the global probability distribution $Q_t$. More training details are provided in the supplementary material.

**Inference Details.** In the inference phase, our method performs multi-stage cross-granularity online optimization. For coarse optimization based on gradient descent, we additionally employ the Adam optimizer with a learning rate $1 \times 10^{-5}$ to finetune the DDT encoder for 100 iterations. For binary search for $\lambda_t$ and $\lambda_c$, the search ranges are set to $[0, 100]$ and $[0, 1]$, respectively.

**Evaluation Protocol.** We evaluate our method on the professional subset in the CLIC validation dataset [36] and Kodak image dataset [19]. The Kodak image dataset contains 24 images with resolutions of $768 \times 512$. The CLIC professional validation dataset comprises 41 images with higher resolutions of about 2K resolutions. The performance is measured by both bit-rates and distortions. We present the bit-rate in bit-per-pixel (bpp) and distortion in Peak Signal-to-Noise Ratio (PSNR). The R-D curves and BD-rate [14] are utilized to compare different methods and settings.

## 4.2. Quantitative Comparison

We compare our method with existing end-to-end learned image compression methods optimized for MSE [11, 15, 24, 25, 40, 46] and conventional compression framework BPG [6] and VVC [8]. Specifically for VVC, we use reference software VTM-12.1 in the evaluation.

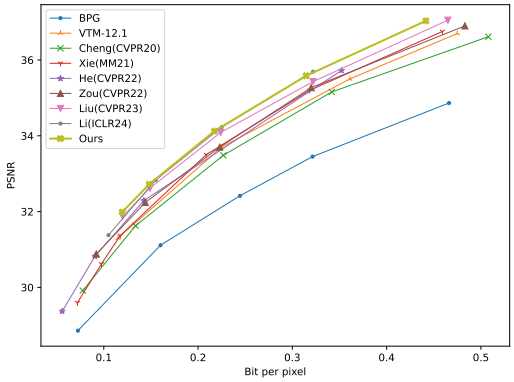


Figure 4. R-D performance evaluated on the CLIC Professional Validation dataset. The compared methods include state-of-the-art LIC models and conventional image codecs.

The R-D curves on CLIC Professional validation sets for our proposed method and comparison methods are shown in Fig. 4. Furthermore, we report the BD-rates [14] results in Table 2 to quantify the average bitrate savings with equal reconstruction quality, with BPG as the anchor. As can be seen, with the addition of cross-granularity online optimization, our approach yields substantial performance gains by more than 2% over our baseline Liu (CVPR-23) [25] with only a 2% increase in parameters. Meanwhile, it also outperforms the latest learned image compression method [24], which has about 60% more parameters than ours. In general, our method achieves state-of-the-art performance and outperforms BPG by 35.21% and 43.86% in BD-rate on the Kodak and CLIC dataset, respectively. The results of Kodak dataset are provided in the supplementary material.

We also provide the complexity analysis of our method in Tab. 1. As a strategy trading time for performance, optimize-based method naturally has a higher encoding time. But we achieve comparable encoding time with widely used VVC [8] and similar decoding time with our baseline Liu (CVPR-23) [25].

Table 1. Time complexity for a 768×512 image.

| Method | Enc. Time | Dec. Time |
|---|---|---|
| VVC [8] | 31.53s | 0.06s |
| Ladune (ICCV-23) [22] | 64.97s | 0.07s |
| Catania (ACMMM-23) [9] | 147.27s | 0.13s |
| Liu (CVPR-23) [25] | 0.25s | 0.24s |
| Ours | 36.34s | 0.25s |

Table 2. BD-rate results and complexity based on CLIC Professional Validation dataset [36]. We set BPG [6] as the anchor in the calculation. The best results are shown in **bold**.

| Method | Param | BD-Rate |
|---|---|---|
| VTM-12.1 [8] | —— | -34.64% |
| Cheng (CVPR-20) [11] | 26.60M | -29.93% |
| Xie (ACMMM-21) [40] | 47.55M | -33.37% |
| He (CVPR-22) [15] | 38.52M | -38.26% |
| Zou (CVPR-22) [46] | 99.58M | -36.84% |
| Liu (CVPR-23) [25] | 42.89M | -41.65% |
| Li (ICLR-24) [24] | 70.97M | -42.80% |
| Ours | 43.76M | **-43.86%** |

### 4.3. Qualitative Comparison

We also compare our method with others in visual quality. The results are shown in Fig. 3. It can be clearly seen that the conventional coding method BPG can reconstruct relatively sharp detailed structures through the modes selection strategy on small coding unit, but its limited flexibility hinders its ability to effectively capture the overall distribution of input samples, leading to noticeable artifacts. Learning-based method Liu (CVPR-23) [25] improves the fitting ability of the overall distribution through the neural network, but it performs poorly on the fine structure because it cannot adequately fit the fine-grained distribution. Our cross-granularity online optimization strategy enhances its fine structure performance while maintaining a better modeling of the overall distribution, and performs better on both fine structure characterization and overall distribution characterization. More qualitative results are shown in the supplementary material.

### 4.4. Ablation Studies

**Effectiveness of DDT and MSC Module.** In our approach, coarse-grained and fine-grained online optimizations are related to the DDT and MSC modules, respectively. Therefore, we first verify the performance of the combination of adaptive QP selection + DDT module and masked selection + MSC module. The results are shown in Fig. 5-(a). w/ DDT means that only DDT module + adaptive QP selection is added, and w/ DDT&MSC means that DDT module + adaptive QP selection and MSC module + masked selection are added at the same time. It can be seen that with the addition of these two combinations, baseline performance are further increased.

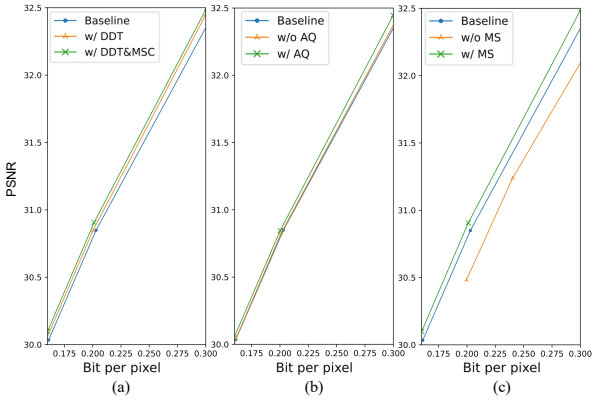


Figure 5. Ablation studies results on the Kodak dataset [19]. [*Zoom in for best view*]

**Effectiveness of Adaptive QP in DDT Module.** Later, we separate the DDT module and the adaptive selection strategy, independently verify the role of the adaptive selection strategy. The result is shown in Fig. 5-(b), w/o AS indicates the result when the adaptive selection strategy is not used, and w/ AS indicates the result when the adaptive selection strategy is used. As we can see, in the absence of an adaptive selection strategy to relax bit rate constraints, it is difficult for the DDT module to learn rich information to improve the overall performance of the model.

**Effectiveness of Mask-based Selection in MSC Module.** Finally, we separate the mask-based selection strategy from the MSC module to verify the role of the mask-based selection strategy. The result is shown in Fig. 5-(c). w/o MS indicates the result when the mask-based selection strategy is not used, and w/ MS indicates the result when the mask-based selection strategy is used. In the case that mask-based selection is not used to extract the sparse representation $y_t$, but all compensation information $y_c$ is transmitted directly, although the compensation information will bring a certain reconstruction quality gain, it is still inferior from the perspective of rate-distortion trade-off because the overall compensation cost is too high.

## 5. Conclusion

In this work, a novel cross-granularity online optimization strategy is proposed to address the amortization effect in learned image compression. In the coarse-grained optimization, gradient descent with adaptive QP minimizes image-level distribution gaps. In the fine-grained level, masked sparse compensation selectively restores structural details under bitrate constraints. Through the joint optimization of global and local distributions, our method achieves better statistical alignment with real image statistics while improving compression performance. Experimental evaluation shows the superiority of our proposed approaches.

## 6. Acknowledgements

This work was supported in part by the Program of Beijing Municipal Science and Technology Commission Foundation under Grant Z241100003524010, and in part by the National Natural Science Foundation of China under Grant 62332010.

## References

[1] Eirikur Agustsson and Radu Timofte. NTIRE 2017 challenge on single image super-resolution: Dataset and study. In *IEEE/CVF Conf. Comput. Vis. Pattern Recog.*, 2017. 7

[2] Johannes Ballé, Valero Laparra, and Eero P Simoncelli. Density modeling of images using a generalized normalization transformation. In *Int. Conf. Learn. Represent.*, 2016. 1, 2

[3] Johannes Ballé, Valero Laparra, and Eero P Simoncelli. End-to-end optimized image compression. In *Int. Conf. Learn. Represent.*, 2017. 2

[4] Johannes Ballé, David Minnen, Saurabh Singh, Sung Jin Hwang, and Nick Johnston. Variational image compression with a scale hyperprior. In *Int. Conf. Learn. Represent.*, 2019. 2

[5] Johannes Ballé, Valero Laparra, and Eero P. Simoncelli. End-to-end optimization of nonlinear transform codes for perceptual quality. In *Picture Coding Symposium*, 2016. 1

[6] Fabrice Bellard. BPG image format. `http://bellard.org/bpg/`, 2017. 1, 7, 8

[7] Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013. 6

[8] Benjamin Bross, Ye-Kui Wang, Yan Ye, Shan Liu, Jianle Chen, Gary J Sullivan, and Jens-Rainer Ohm. Overview of the versatile video coding (VVC) standard and its applications. *IEEE Trans. Circuit Syst. Video Technol.*, 31(10): 3736–3764, 2021. 3, 7, 8

[9] Lorenzo Catania and Dario Allegra. NIF: A fast implicit image compression with bottleneck layers and modulated sinusoidal activations. In *ACM Int. Conf. Multimedia*, 2023. 3, 8

[10] Tong Chen, Haojie Liu, Zhan Ma, Qiu Shen, Xun Cao, and Yao Wang. End-to-end learnt image compression via non-local attention optimization and improved context modeling. *IEEE Trans. Image Process.*, 30:3179–3191, 2021. 2

[11] Zhengxue Cheng, Heming Sun, Masaru Takeuchi, and Jiro Katto. Learned image compression with discretized gaussian mixture likelihoods and attention modules. In *IEEE/CVF Conf. Comput. Vis. Pattern Recog.*, 2020. 1, 2, 7, 8

[12] Chris Cremer, Xuechen Li, and David Duvenaud. Inference suboptimality in variational autoencoders. *Int. Conf. Mach. Learn.*, 2018. 2, 3

[13] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *IEEE/CVF Conf. Comput. Vis. Pattern Recog.*, 2021. 6

[14] Bjøntegaard Gisle. Calculation of average PSNR differences between RD curves. In *VCEG-M33*, 2001. 7

[15] Dailan He, Ziming Yang, Weikun Peng, Rui Ma, Hongwei Qin, and Yan Wang. ELIC: Efficient learned image compression with unevenly grouped space-channel contextual adaptive coding. In *IEEE/CVF Conf. Comput. Vis. Pattern Recog.*, 2022. 1, 7, 8

[16] Wei Jiang, Wei Wang, and Yue Chen. Neural image compression using masked sparse visual representation. In *IEEE/CVF Winter Conf. Appli. Comput. Vis.*, 2024. 2

[17] Yoon Kim, Sam Wiseman, Andrew Miller, David Sontag, and Alexander Rush. Semi-amortized variational autoencoders. In *Int. Conf. Mach. Learn.*, 2018. 3

[18] Diederik Pieter Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Int. Conf. Learn. Represent.*, 2014. 7

[19] Eastman Kodak. Kodak lossless true color image suite. `https://r0k.us/graphics/kodak/`, 2013. 7, 8

[20] A. Burakhan Koyuncu, Han Gao, Atanas Boev, Georgii Gaikov, Elena Alshina, and Eckehard Steinbach. Contextformer: A transformer with spatio-channel attention for context modeling in learned image compression. In *Eur. Conf. Comput. Vis.*, 2022. 2

[21] Haowei Kuang, Yiyang Ma, Wenhan Yang, Zongming Guo, and Jiaying Liu. Consistency guided diffusion model with neural syntax for perceptual image compression. In *ACM Int. Conf. Multimedia*, 2024. 5

[22] Théo Ladune, Pierrick Philippe, Félix Henry, Gordon Clare, and Thomas Leguay. COOL-CHIC: Coordinate-based low complexity hierarchical image codec. In *Int. Conf. Comput. Vis.*, 2023. 3, 8

[23] Jooyoung Lee, Seunghyun Cho, and Seung-Kwon Beack. Context-adaptive entropy model for end-to-end optimized image compression. In *Int. Conf. Learn. Represent.*, 2018. 2

[24] Han Li, Shaohui Li, Wenrui Dai, Chenglin Li, Junni Zou, and Hongkai Xiong. Frequency-aware transformer for learned image compression. In *Int. Conf. Learn. Represent.*, 2024. 2, 7, 8

[25] Jinming Liu, Heming Sun, and Jiro Katto. Learned image compression with mixed transformer-CNN architectures. In *IEEE/CVF Conf. Comput. Vis. Pattern Recog.*, 2023. 1, 2, 4, 7, 8

[26] Ming Lu, Peiyao Guo, Huiqing Shi, Chuntong Cao, and Zhan Ma. Transformer-based image compression. In *Data Compression Conf.*, 2022. 2

[27] Qi Mao, Tinghan Yang, Yinuo Zhang, Zijian Wang, Meng Wang, Shiqi Wang, Libiao Jin, and Siwei Ma. Extreme image compression using fine-tuned VQGANs. In *Data Compression Conf.*, 2024. 2

[28] Michael W Marcellin, Michael J Gormish, Ali Bilgin, and Martin P Boliek. An overview of JPEG-2000. In *Data Compression Conf.*, 2000. 1

[29] David Minnen and Saurabh Singh. Channel-wise autoregressive entropy models for learned image compression. In *IEEE Int. Conf. Image Process.*, 2020. 2

[30] David Minnen, George Toderici, Michele Covell, Troy Chinen, Nick Johnston, Joel Shor, Sung Jin Hwang, Damien

Vincent, and Saurabh Singh. Spatially adaptive image compression using a tiled deep network. In *IEEE Int. Conf. Image Process.*, 2017. 2

[31] David Minnen, Johannes Ballé, and George D Toderici. Joint autoregressive and hierarchical priors for learned image compression. In *Adv. Neural Inform. Process. Syst.*, 2018. 2

[32] David Minnen, George Toderici, Saurabh Singh, Sung Jin Hwang, and Michele Covell. Image-dependent local entropy models for learned image compression. In *IEEE Int. Conf. Image Process.*, 2018. 2

[33] Claude E Shannon. Coding theorems for a discrete source with a fidelity criterion. *IRE Nat. Conv. Rec*, 4:142–163, 1959. 3

[34] Yannick Strümpler, Janis Postels, Ren Yang, Luc Van Gool, and Federico Tombari. Implicit neural representations for image compression. In *Eur. Conf. Comput. Vis.*, 2022. 3

[35] George Toderici, Damien Vincent, Nick Johnston, Sung Jin Hwang, David Minnen, Joel Shor, and Michele Covell. Full resolution image compression with recurrent neural networks. In *IEEE/CVF Conf. Comput. Vis. Pattern Recog.*, 2017. 1

[36] George Toderici, Wenzhe Shi, Radu Timofte, Lucas Theis, Johannes Balle, Eirikur Agustsson, Nick Johnston, and Fabian Mentzer. Workshop and challenge on learned image compression. In *IEEE/CVF Conf. Comput. Vis. Pattern Recog. Worksh.*, 2020. 7, 8

[37] Gregory K Wallace. The JPEG still picture compression standard. *Commun. ACM*, 34(4):30–44, 1991. 1

[38] Dezhao Wang, Wenhan Yang, Yueyu Hu, and Jiaying Liu. Neural data-dependent transform for learned image compression. In *IEEE/CVF Conf. Comput. Vis. Pattern Recog.*, 2022. 2, 3, 5

[39] Yefei Wang, Dong Liu, Siwei Ma, Feng Wu, and Wen Gao. Ensemble learning-based rate-distortion optimization for end-to-end image compression. *IEEE Trans. Circuit Syst. Video Technol.*, 31(3):1193–1207, 2020. 2, 3

[40] Yueqi Xie, Ka Leong Cheng, and Qifeng Chen. Enhanced invertible encoding for learned image compression. In *ACM Int. Conf. Multimedia*, 2021. 1, 7, 8

[41] Naifu Xue, Qi Mao, Zijian Wang, Yuan Zhang, and Siwei Ma. Unifying generation and compression: Ultra-low bitrate image coding via multi-stage transformer. In *Int. Conf. Multimedia and Expo*, 2024. 2

[42] Yibo Yang, Robert Bamler, and Stephan Mandt. Improving inference for neural image compression. In *Adv. Neural Inform. Process. Syst.*, 2020. 2

[43] Haotian Zhang and Dong Liu. The gap between principle and practice of lossy image coding. *arXiv preprint arXiv:2501.12330*, 2025. 3

[44] Yunfei Zheng, Peng Yin, Qian Xu, Joel Sole, and Xiaoan Lu. Directional adaptive loop filter for video coding. In *IEEE Int. Conf. Image Process.*, 2011. 2

[45] Yinhao Zhu, Yang Yang, and Taco Cohen. Transformer-based transform coding. In *Int. Conf. Learn. Represent.*, 2022. 1

[46] Renjie Zou, Chunfeng Song, and Zhaoxiang Zhang. The devil is in the details: Window-based attention for image compression. In *IEEE/CVF Conf. Comput. Vis. Pattern Recog.*, 2022. 1, 7, 8