# Knowledge Distillation for Learned Image Compression

Yunuo Chen[1]* Zezheng Lyu[2]* Bing He[1] Ning Cao[3] Gang Chen[3] Guo Lu[1] ✉ Wenjun Zhang[1]
[1] Shanghai Jiao Tong Unversity [2] Massachusetts Institute of Technology
[3] E-surfing Vision Technology Co., Ltd.

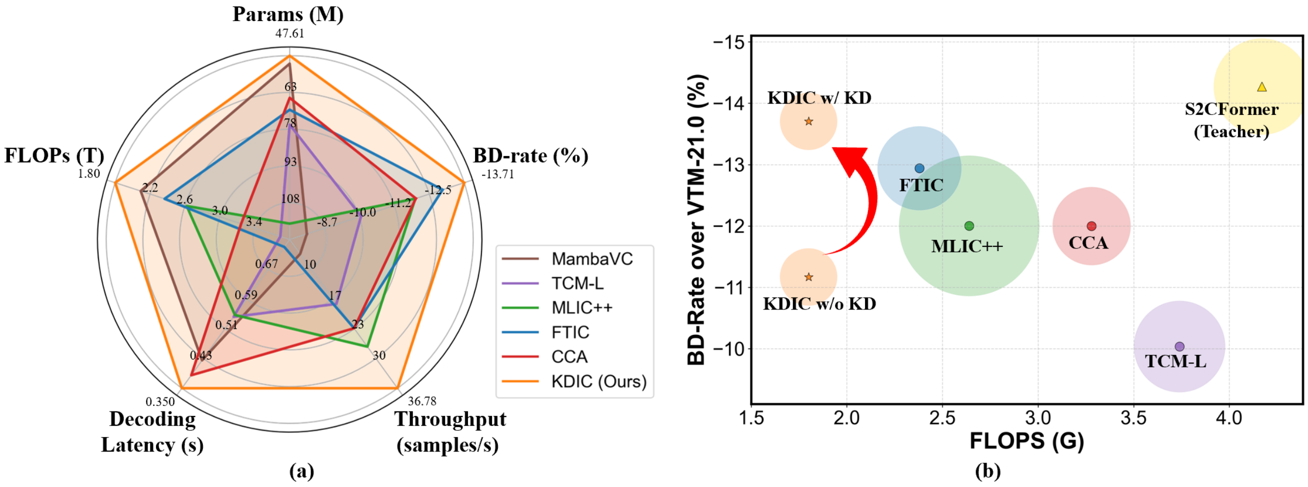


Figure 1. **State-of-the-Art Performance of KDIC.** We propose a knowledge distillation framework for learned image compression and develop KDIC, a compact and efficient student model based on S2CFormer [12]. As shown in Figure (a), KDIC outperforms recent advanced methods in rate-distortion (RD) metrics and model complexity measures. Figure (b) highlights the effectiveness of knowledge distillation: it achieves a 2.5% reduction in BDrate while reducing parameters by 40% and FLOPs by 57% compared to the teacher model. The circle radius corresponds to the parameter count of each model.

## Abstract

*Recently, learned image compression (LIC) models have achieved remarkable rate-distortion (RD) performance, yet their high computational complexity severely limits practical deployment. To overcome this challenge, we propose a novel **S**tage-wise **Mo**dular **Di**stillation framework, **SMoDi**, which efficiently compresses LIC models while preserving RD performance. This framework treats each stage of LIC models as an independent sub-task, mirroring the teacher model's task decomposition to the student, thereby simplifying knowledge transfer. We identify two crucial factors determining the effectiveness of knowledge distillation: student model construction and loss function design. Specifically, we first propose Teacher-Guided Student Model Construction, a pruning-like method ensuring architectural consistency between teacher and student models. Next, we introduce Implicit End-to-end Supervision, facilitating adaptive energy compaction and bitrate regularization. Based on these insights, we develop **KDIC**, a lightweight student model derived from the state-of-the-art S2CFormer model. Experimental results demonstrate that KDIC achieves top-tier RD performance with significantly reduced computational complexity. To our knowledge, this work is among the first successful applications of knowledge distillation to learned image compression.*

## 1. Introduction

The rapid growth of digital image data demands efficient compression methods. Deep neural network–based learned image compression (LIC) has recently delivered significant advances in rate-distortion (RD) performance compared to traditional compression methods, achieving enhanced compression efficiency and storage savings [3, 4, 26, 45, 46, 63].

∗ Equal Contribution ✉ Corresponding Author

Despite these benefits, existing LIC models often require high computational resources, resulting in a suboptimal tradeoff between performance and complexity. For example, as shown in Fig. 1 (a), MLIC++ [25] suffers from excessive parameters, FTIC [33] faces high decoding latency, and TCM-L [40] requires substantial floating-point operations.

To overcome the previous tradeoff between rate-distortion (RD) performance and model complexity, we turn to knowledge distillation. This technique transfers capabilities from larger, high-performing teacher models to more compact student models[1, 24, 50, 56, 57]. Although this approach has been applied in various domains, its dedicated application in LIC remains underexplored. Moreover, applying knowledge distillation to LIC models poses unique challenges inherent to compression tasks.

We propose **S**tage-wise **Mo**dular **Di**stillation framework, **SMoDi**, a novel knowledge distillation framework tailored for LIC. Given the unique requirements of image compression, our framework treats each stage as an independent sub-task. This approach decomposes the complex overall task handled by the teacher model into simpler sub-tasks for the student model, thereby simplifying knowledge transfer. Next, We identify that the efficacy of knowledge distillation in LIC fundamentally depends on two critical factors: the architectural design of student-teacher model pairs and the formulation of appropriate distillation loss functions.

Firstly, architectural mismatches are a common issue for low-level tasks and knowledge transfer. Variations in model architectures (e.g., convolution-based, transformer-based, or mamba-based structures) can result in the capacity gap that hinders knowledge transfer [9, 47]. We propose Teacher-Guided Student Model Construction, a pruning-like method that builds the student model based on the teacher model, ensuring structural consistency and facilitating effective knowledge transfer. The second challenge lies in the design of the loss function, which must simultaneously account for energy biases and bitrate regularization. Intermediate features in LIC models tend to exhibit pronounced energy compaction, where most energy is concentrated in a few channels, leading to disparities in channel importance and information content. Besides, an inappropriate fluctuation in energy distribution can significantly affect the bit rate. To overcome this, we introduce Implicit End-to-end Supervision, which enables intermediate layers to adaptively learn appropriate energy compaction patterns, thus robustly managing reconstruction quality and bitrate. This strategy also mitigates challenges related to discrepancies in channel count and feature sizes during explicit intermediate feature supervision.

By integrating these alignment mechanisms with our stage-wise distillation framework, we systematically overcome the primary constraints hindering effective knowledge distillation in LIC models. To demonstrate the effectiveness of our approach, we apply our framework to the state-of-the-art S2CFormer model and develop **KDIC**, a compact, lightweight student model. KDIC demonstrates superior RD performance and efficiency compared to previous methods, showcasing the practical benefits of our framework.

Our main contributions are summarized as follows:

- We propose **SMoDi**, a modular knowledge distillation framework explicitly tailored to LIC tasks, effectively transferring knowledge from complex teacher models to compact student architectures.
- We identify two key barriers to knowledge distillation in LIC models: architectural mismatches and supervision inefficiency. We propose Teacher-Guided Student Model Construction for architecture alignment and Implicit End-to-end Supervision for adaptive energy compaction and bitrate regularization to address these issues.
- We implement and validate **KDIC**, a student model distilled from the state-of-the-art S2CFormer architecture, demonstrating superior RD performance with significantly reduced complexity compared to existing LIC methods.

## 2. Related Work

### 2.1. Learned Image Compression

Recent years have witnessed a surge in end-to-end learned image compression (LIC), spearheaded by Ballé *et al.*[3, 4]. Their work, which leverages convolutional neural networks and variational auto-encoders with hyper-prior frameworks, has been a cornerstone for subsequent research in end-to-end learned image compression. Subsequent research has focused on enhancing rate-distortion performance along two primary avenues: improved transform networks and advanced entropy models [5, 16, 17, 32, 39, 42–44, 51–53, 58, 62].

Regarding transform networks, innovations range from the use of invertible neural networks [55] and residual blocks [13] to octave residual modules [10]. Moreover, researchers have integrated transformer architectures—exemplified by Swin Transformers [64, 65], Resblock-Swin hybrids [40], and frequency-driven window attention mechanisms [33]—to further boost performance. In parallel, entropy modeling has evolved significantly. Early methods based on autoregressive schemes [46] and checkerboard patterns [20] have given way to more sophisticated approaches such as channel-dimension context models [45] and uneven space-channel adaptive coding [21]. Recent advances also include quadtree models [35, 36], ViT-based context enhancement [48], multi-reference models [25, 26], and transformer-based channel-wise autoregression [33].

Despite these impressive advances, the complexity of current LIC models remains a significant obstacle to practical deployment. Metrics such as parameter count, FLOPs, decoding latency, and training throughput indicate that these models are computationally intensive. This high complexity increases resource requirements and poses challenges for real-time applications. Consequently, a growing interest is in exploring more efficient architectures that balance compression accuracy with reduced computational demands. Accordingly, this paper focuses on reducing model complexity while preserving compression performance, enabling broader practical use of LIC models.

## 2.2. Knowledge Distillation

Knowledge distillation has become one of the most effective model compression and acceleration techniques. Its primary goal is to transfer the rich representational capacity of a large model (the teacher) to a smaller one (the student), thereby enhancing the student's performance. Initially introduced for classification tasks [1, 23, 24, 50, 56, 57, 59], the method utilizes the teacher's softmax outputs as soft labels, offering more nuanced guidance than traditional complex labels. Building on this foundation, researchers have extended the approach to feature distillation, where intermediate feature maps from the teacher are transferred to the student [27, 41, 61]. In these methods, manually designed transformations align the teacher's and student's feature representations [30, 61]. Additionally, self-distillation techniques have been proposed to transfer knowledge from deeper to shallower layers within the same network [38, 60]. Beyond image classification, knowledge distillation has also proven effective in domains such as object detection and image segmentation [41, 54], as well as in low-level tasks like super-resolution [18, 22, 31].

However, conventional distillation strategies face challenges when applied to LIC models. The unique VAE-based structure and the energy compaction phenomenon inherent to compression models complicate the direct use of standard techniques [15, 21, 34, 37]. Consequently, there is a pressing need to develop specialized distillation methods that effectively address these distinct characteristics.

# 3. Method

## 3.1. Preliminary

Fig. 2 illustrates the Teacher-Student LIC framework, which comprises three main components: an encoder, a decoder, and an entropy model. Starting with an input RGB image $\boldsymbol{x}$, the encoder $g_a(\cdot)$ generates a latent representation $\boldsymbol{y}$. This latent is quantized to obtain $\hat{\boldsymbol{y}}$. The decoder $g_s(\cdot)$ then reconstructs the image as $\hat{\boldsymbol{x}}$ from $\hat{\boldsymbol{y}}$. Concurrently, a hyper encoder $h_a(\cdot)$ transforms $\boldsymbol{y}$ into a hyper-latent $\boldsymbol{z}$, which is also quantized into $\hat{\boldsymbol{z}}$. A hyper decoder $h_s(\cdot)$ uses $\hat{\boldsymbol{z}}$ to compute the Gaussian parameters $(\boldsymbol{\mu}, \boldsymbol{\sigma})$ governing the distribution of $\hat{\boldsymbol{y}}$. These processes can be expressed as:

$$\begin{aligned} &\boldsymbol{y} = g_a(\boldsymbol{x}; \boldsymbol{\theta}_a), \quad \hat{\boldsymbol{y}} = Q(\boldsymbol{y} - \boldsymbol{\mu}) + \boldsymbol{\mu}, \quad \hat{\boldsymbol{x}} = g_s(\hat{\boldsymbol{y}}; \boldsymbol{\theta}_s), \\ &\boldsymbol{z} = h_a(\boldsymbol{y}; \boldsymbol{\phi}_a), \quad \hat{\boldsymbol{z}} = Q(\boldsymbol{z}), \quad \boldsymbol{\mu}, \boldsymbol{\sigma} = h_s(\hat{\boldsymbol{z}}; \boldsymbol{\phi}_s). \end{aligned}$$

To balance compression efficiency and reconstruction fidelity, we introduce an objective function $L$ defined by:

$$L = R + \lambda D,$$

where the trade-off parameter $\lambda$ controls the importance of distortion relative to bitrate. Bitrate and distortion are defined as

$$\begin{aligned} R &= \mathbb{E}\Big[-\log_2 p_{\hat{\boldsymbol{y}}|\hat{\boldsymbol{z}}}(\hat{\boldsymbol{y}} \mid \hat{\boldsymbol{z}})\Big] + \mathbb{E}\Big[-\log_2 p_{\hat{\boldsymbol{z}}}(\hat{\boldsymbol{z}})\Big], \\ D &= \mathbb{E}\Big[\|\boldsymbol{x} - \hat{\boldsymbol{x}}\|_2^2\Big]. \end{aligned}$$

The encoding and decoding processes in LIC are each divided into three stages, with divisions occurring at nodes where feature resolution changes due to upsampling or downsampling operations. This creates a total of six distinct stages. Each stage comprises one downsampling/upsampling module and multiple non-linear transform blocks. We approach each stage as an independent sub-task and perform distillation in a per-stage manner. This stage-wise decomposition is the basis for the efficient transfer of knowledge from the teacher model to the student model.

## 3.2. Stage-wise Modular Distillation

### 3.2.1. Distillation for Non-linear Transforms

This section introduces our Stage-wise Modular Distillation framework, SMoDi, designed to systematically transfer knowledge from a complex teacher model to a more efficient student model. As shown in Fig. 2, the teacher model's overall task is decomposed into a series of independent sub-tasks, each representing a specific network stage. This framework systematically breaks down the complex tasks executed by the teacher into simpler, well-defined sub-tasks for the student to learn and mimic. Each sub-task is assigned to a corresponding module in the student model, fostering a modular and efficient knowledge transfer. Each module in the student model is trained to master its assigned sub-task by mirroring the behavior of the corresponding stage in the teacher model. Each module focuses on a specific aspect by breaking the task into manageable sub-tasks, reducing task complexity and ambiguity and simplifying learning.

Conventional knowledge distillation methods typically rely on end-to-end training with intermediate feature supervision, where each stage is initialized and trained together from scratch. This approach can lead to several challenges: during forward propagation, the incompletely trained parameters and noise in the early stages can propagate errors to subsequent stages, causing instability in the training process. During backpropagation, the gradients may suffer from inaccuracies or degradation as they are passed through multiple incompletely trained stages, further exacerbating the training inefficiency. In our framework, however, each stage of the student model is integrated into a pre-trained and

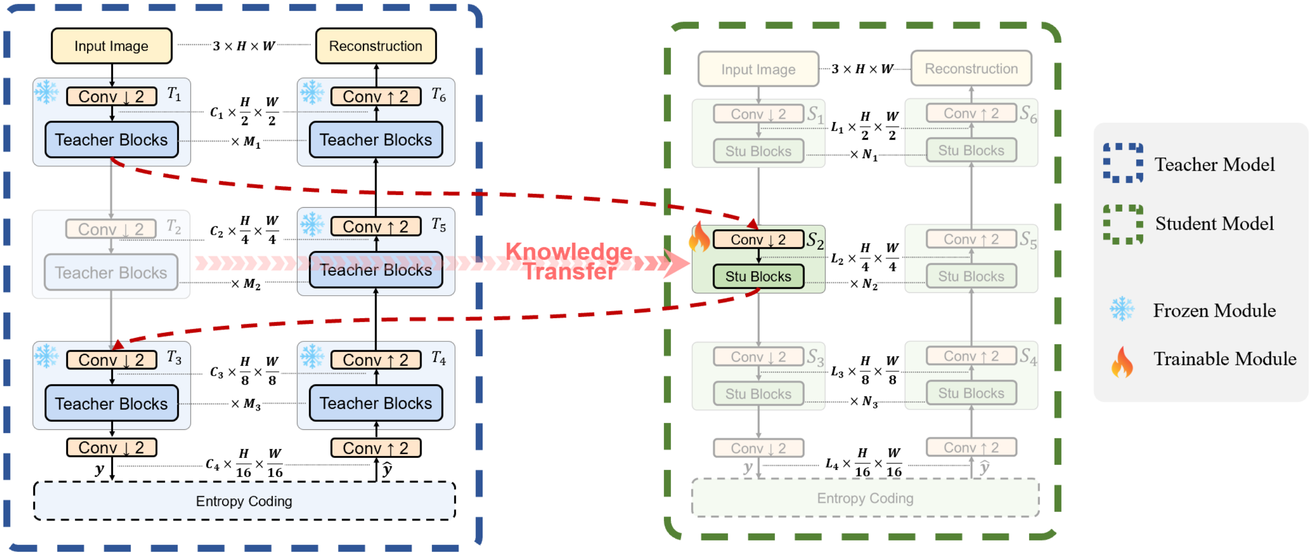


Figure 2. **Overview of our SMoDi framework.** The encoding and decoding processes are each split into 3 stages at nodes where feature resolution shifts via upsampling (US) or downsampling (DS). This results in 6 distinct stages overall. Each stage consists of one DS/US module and several non-linear transform blocks. In the teacher model, the stages are labeled $T_1$–$T_6$, with channel configurations $C_1$–$C_6$ and block counts $M_1$–$M_6$. Similarly, in the student model, stages $S_1$–$S_6$ have channel configurations $L_1$–$L_6$ and block counts $N_1$–$N_6$. The knowledge from the teacher model is effectively transferred to the student model in a stage-wise manner.

frozen teacher model, which allows the student modules to be trained more precisely and efficiently.

We also provide some theoretical insights to demonstrate the advantages of the stage-wise approach over conventional methods later. As our SMoDi framework splits the LIC primary transform into stages, we can rewrite the encoding and decoding processes as follows:

$$\begin{aligned}
\boldsymbol{y}^T &= T_a(\boldsymbol{x}; \boldsymbol{\Theta}_a) = T_3(\boldsymbol{\Theta}_3) \circ T_2(\boldsymbol{\Theta}_2) \circ T_1(\boldsymbol{\Theta}_1)(\boldsymbol{x}), \\
\hat{\boldsymbol{x}}^T &= T_s(\hat{\boldsymbol{y}}; \boldsymbol{\Theta}_s) = T_6(\boldsymbol{\Theta}_6) \circ T_5(\boldsymbol{\Theta}_5) \circ T_4(\boldsymbol{\Theta}_4)(\hat{\boldsymbol{y}}), \\
\boldsymbol{y}^S &= S_a(\boldsymbol{x}; \boldsymbol{\Psi}_a) = S_3(\boldsymbol{\Psi}_3) \circ S_2(\boldsymbol{\Psi}_2) \circ S_1(\boldsymbol{\Psi}_1)(\boldsymbol{x}), \\
\hat{\boldsymbol{x}}^S &= S_s(\hat{\boldsymbol{y}}; \boldsymbol{\Psi}_s) = S_6(\boldsymbol{\Psi}_6) \circ S_5(\boldsymbol{\Psi}_5) \circ S_4(\boldsymbol{\Psi}_4)(\hat{\boldsymbol{y}}),
\end{aligned}$$

where $T_i$ ($S_i$) denotes the $i$-th block of the teacher model (student model), parameterized by $\boldsymbol{\Theta}_i$ ($\boldsymbol{\Psi}_i$).

Let $P(\boldsymbol{x})$ denote the distribution of the input raw image. For the purpose of providing preliminary theoretical insights, here we assume each block's irreversibility and differentiability. Then, for the latent representation distributions, we have:

$$P(\boldsymbol{y}^T|\boldsymbol{\Theta}_a) = P(\boldsymbol{x}) \cdot J_{T_a}(x)^{-1},$$

$$P(\boldsymbol{y}^S|\boldsymbol{\Psi}_a) = P(\boldsymbol{x}) \cdot J_{S_a}(x)^{-1},$$

where $J$ denotes the absolute Jacobian determinant.

In distillation, our goal is for $P(\boldsymbol{y}^S|\boldsymbol{\Psi}_a)$ to approximate $P(\boldsymbol{y}^T|\boldsymbol{\Theta}_a)$ as closely as possible. However, since each block is parameterized by learned parameters (e.g., $\boldsymbol{\Psi}_i$ for $S_i$), which are optimized using methods like Adam, and due to training randomness (e.g., uniform noise, parameter initialization, and batch sampling), each Jacobian determinant becomes stochastic, introducing variance.

Intuitively, when blocks are trained stage-wise, each stage learns separately, keeping training errors uncorrelated. This keeps errors small, localized, and independent, maintaining a stable latent distribution close to the teacher's. In contrast, joint training trains layers together from scratch, causing errors to correlate and amplify randomness across layers. This accumulates fluctuations, causing the latent distribution to deviate further from the stable teacher model. Thus, the stage-wise method is expected to perform better.

To mathematically justify this, let $M_1$ denote the stage-wise method, $M_2$ denote the joint training method, $P_T(\boldsymbol{y}) = P(\boldsymbol{y}^T|\boldsymbol{\Theta}_a)$, and $P_S(\boldsymbol{y}) = P(\boldsymbol{y}^S|\boldsymbol{\Psi}_a)$. We measure the difference between the two distributions using the KL divergence. We can roughly assume that (1) determinant of each block's Jacobian follows an invariant marginal distribution with fixed mean and variance regardless of the training method, and is unbiased about the teacher model's target block; (2) in joint training, dependencies increase each block's co-movement in magnitudes but do not increase the absolute mean of each block's product. Then, we state that:

$$\frac{D_{\mathrm{KL}}(P_T \| P_S)_{M_2}}{D_{\mathrm{KL}}(P_T \| P_S)_{M_1}} \approx \frac{\mathrm{Var}(J_{S_a}^{-1}(x))_{M_2}}{\mathrm{Var}(J_{S_a}^{-1}(x))_{M_1}} \geq 1.$$

For detailed proof and discussion of our findings, we refer the reader to the appendix.

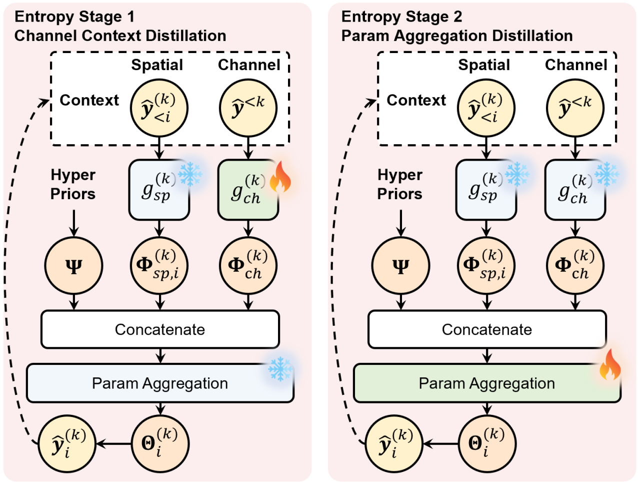

Entropy Stage 1
Channel Context Distillation
Spatial
Channel
Context
$\hat{\boldsymbol{y}}^{(k)}_{<i}$
$\hat{\boldsymbol{y}}^{<k}$
Hyper Priors
$g^{(k)}_{sp}$
$g^{(k)}_{ch}$
$\boldsymbol{\Psi}$
$\boldsymbol{\Phi}^{(k)}_{sp,i}$
$\boldsymbol{\Phi}^{(k)}_{ch}$
Concatenate
Param Aggregation
$\hat{\boldsymbol{y}}^{(k)}_{i}$
$\boldsymbol{\Theta}^{(k)}_{i}$
Entropy Stage 2
Param Aggregation Distillation
Spatial
Channel
Context
$\hat{\boldsymbol{y}}^{(k)}_{<i}$
$\hat{\boldsymbol{y}}^{<k}$
Hyper Priors
$g^{(k)}_{sp}$
$g^{(k)}_{ch}$
$\boldsymbol{\Psi}$
$\boldsymbol{\Phi}^{(k)}_{sp,i}$
$\boldsymbol{\Phi}^{(k)}_{ch}$
Concatenate
Param Aggregation
$\hat{\boldsymbol{y}}^{(k)}_{i}$
$\boldsymbol{\Theta}^{(k)}_{i}$

Figure 3. The two-stage distillation for the entropy model. We distill two main parts of SCCTX [21] entropy model by stages: Channel Context and Parameter Aggregation modules.

### 3.2.2. Distillation for Entropy Model

For LIC models, the entropy model—alongside the non-linear transforms—is also a pivotal component. We apply the stage-wise modular distillation (SMoDi) strategy to the entropy model as well, decomposing its structure and tasks to achieve more effective knowledge transfer.

An SCCTX entropy model [21] predicts Gaussian parameters $\Theta_i^{(k)} = (\mu, \sigma)$ from three contexts:

$$\Phi_{\mathrm{sp},i}^{(k)} = g_{\mathrm{sp}}^{(k)}\big(\hat{\mathbf{y}}_{<i}^{(k)}\big), \quad \Phi_{\mathrm{ch}}^{(k)} = g_{\mathrm{ch}}^{(k)}\big(\hat{\mathbf{y}}^{(<k)}\big), \quad \Psi = g_{\mathrm{hyper}}(\hat{\mathbf{z}}),$$

followed by an *Aggregation* network:

$$\Theta_i^{(k)} = \mathrm{Agg}\big(\Phi_{\mathrm{sp},i}^{(k)}, \Phi_{\mathrm{ch}}^{(k)}, \Psi\big).$$

**Distillation principle.** We identify that Channel-context module and Parameter-aggregation module are two main parts containing large amounts of parameters. We sequentially transfer these two learnable blocks while freezing the rest of the student's entropy path:

1. *Channel-context stage.* Unfreeze $g_{\mathrm{ch}}^{(k)}$ only and let student's channel context modules mimic the teacher's $\Phi_{\mathrm{ch}}^{(k)}$.
2. *Parameter-aggregation stage.* Freeze $g_{\mathrm{ch}}^{(k)}$ and others, unfreeze aggregation networks to learn from teacher's parameter aggregation.

This two-step schedule for entropy model also isolates the sub-tasks, stabilises gradients, and yields better rate-distortion performance than distilling the whole entropy model at once.

## 3.3. Teacher-Guided Student Model Construction

The performance of a student model in knowledge distillation depends on both the teacher model's efficacy and the student's ability to mimic the teacher [9]. Specifically, when the capacity gap between teacher and student is too large, the student struggles to mimic effectively, resulting in suboptimal distillation. Consequently, it is crucial to respect this transfer gap in knowledge distillation [47]. In image compression, where pixel-level precision is essential, the architectural distinctions (e.g., between CNNs, Transformers, and Mambas) are particularly magnified. For example, a convolution-based network may struggle to mimic a transformer-based teacher due to significant disparities in their feature representations. Therefore, maintaining structural consistency between teacher and student models is critical for effective knowledge transfer in LIC.

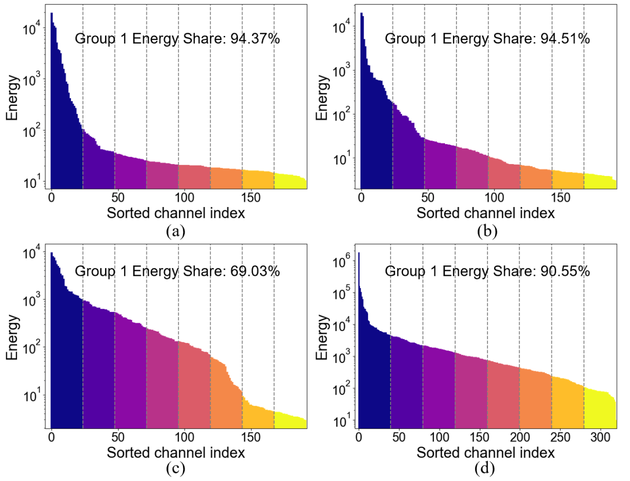


Figure 4. Channel-wise energy distribution of S2CFormer's intermediate features [12], averaged over 24 images from the Kodak dataset. The latent channels are sorted into eight groups. Panels (a)–(c) display histograms for the features produced by the first three stages, while panel (d) presents the histogram for the final latent representation. Notably, the first group—capturing low-frequency features—accounts for an average of 87% of the total energy, whereas the remaining groups, corresponding to high-frequency features, exhibit significantly lower energy.

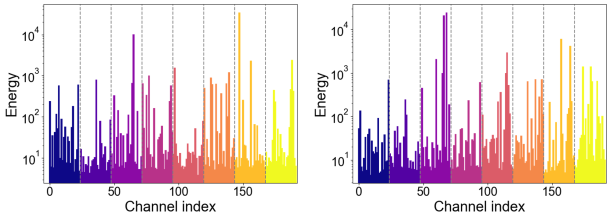
We propose constructing the student model by pruning the teacher model to address this issue. This approach involves selectively removing redundant components from the teacher model to reduce computational complexity while preserving its core performance. Specifically, the model is simplified by reducing the number of blocks, decreasing the channel count, and lowering the FeedForward network (FFN) expansion factor. By using this teacher-guided pruning strategy, the student model inherits the teacher's essential capabilities while achieving lower resource consumption. This method not only avoids the pitfalls of cross-architecture distillation but also ensures that the distilled model remains both efficient and effective. The experimental section provides a detailed description and comparative analysis of this approach.

### 3.4. Implicit End-to-end Supervision

Another key deterministic component in knowledge distillation is the design of loss functions. Traditional knowledge distillation methods fall into two categories: logit distillation and feature distillation. Logit distillation aligns teacher and student outputs using soft labels (e.g., via KL divergence) but appears unsuitable for low-level tasks like image compression, which lack categorical information. Feature distillation, on the other hand, aligns intermediate features (e.g., via L2 loss) but faces significant challenges in LIC due to the uneven energy distribution across channels.

As shown in Fig.4, LIC models exhibit channel-wise energy compaction: most of the energy is concentrated in a few channels, leading to substantial variations in channel importance and information content. This energy concentration phenomenon is related to bitrate regularization. Different channel groups convey distinct frequency information, so inappropriate fluctuations in these unevenly distributed channels can lead to drastic bitrate changes. LIC models must address reconstruction fidelity, feature consistency, and bitrate penalty. Supervising feature similarity alone does not adequately regulate the bitrate, causing it to increase undesirably. Traditional feature distillation treats all channels equally, neglecting this energy imbalance. Moreover, when teacher and student models have mismatched channel numbers, the rearrangement and re-compaction of energy further complicate conventional channel adaptation methods.

To overcome these challenges, we propose Implicit End-to-end Supervision, a novel approach tailored to the specific demands of LIC. Instead of enforcing explicit channel-wise feature alignment, our method directly integrates the end-to-end RD loss for supervision. This strategy enables the student model to implicitly learn effective energy compaction patterns and align with the teacher's energy distribution while ensuring that the bitrate remains well-regulated. Thus, our approach simplifies the overall supervision process, adapts seamlessly to diverse channel configurations, and robustly manages reconstruction quality and bitrate. We demonstrate the channel-wise energy of teacher and student features (Fig. 5) to show that implicit supervision allows the student block to reallocate and re-compacts energy across channels. This redistribution indicates that implicit supervision can enable more flexible energy compaction.



(a) Teacher Block

(b) Student Block

Figure 5. Unsorted Channel-wise Energy Distribution.

# 4. Experiments

## 4.1. Experimental Setup

### 4.1.1. Training Details

Models are trained on the Flickr2W [39] with Adam [28] optimizer and an initial learning rate of 0.0001. For MSE-optimized models, we use Lagrangian multipliers $\{0.0017, 0.0025, 0.0035, 0.0067, 0.0130, 0.0250, 0.050\}$. The models are trained 0.25 million steps for stage-wise modular knowledge transfer and finetuned end-to-end for 0.75 million steps. Experiments are conducted on NVIDIA A100 GPUs.

### 4.1.2. Evaluation

We test our models on three datasets: Kodak [29] with the image size of 768 × 512, Tecnick [2] with the image size of 1200 × 1200, and CLIC professional validation dataset [14] with 2k resolution. For RD performance, we take PSNR and bits per pixel (bpp) as metrics. We utilize the BD-rate [6] to quantify the average bitrate savings. We also take Parameter Count, FLOPs, Decoding Latency and Training Throughput as metrics for model complexity. **Note:** PSNR is calculated on RGB and all the latency is calculated on 2K images.

### 4.1.3. Model Details

We choose S2CFormer-Hybrid [12] as the teacher model. For S2CFormer-Hybrid, channel numbers of intermediate features $\{C_1, C_2, C_3, C_4\}$ are set as $\{192, 192, 192, 320\}$ and the numbers of non-linear transform blocks $\{L_1, L_2, L_3\}$ are set as $\{3, 8, 8\}$. In our design of KDIC, we prune S2CFormer by reducing the channel number of the first stage and the block number of the next two stages. So, for KDIC, we set $\{128, 192, 192, 320\}$ for channels and $\{3, 5, 5\}$ for block numbers. Besides, we also shrink the expansion factors in FFNs from 4 to 2.

## 4.2. Performance of KDIC.

In this section, we compare our model with recent advanced traditional and learned methods, including VTM-21.0 [7, 8], ELIC [21], TCM-L [40], MLIC++ [25], FTIC [33] and CCA [19]. We make comprehensive comparisons, including RD performance on different resolution image datasets and the complexity metrics. Detailed results are provided in Tab.1.

**R-D Performance.** As shown in Tab.1, KDIC outperforms VTM-21.0 by 13.71%, 12.63% and 16.64% in BD-rate on the Kodak, CLIC and Tecnick datasets, respectively. RD curves of KDIC and other advanced methods are provided in Fig.6-8. Our KDIC model achieves SOTA RD performance in terms of PSNR BD-rate, across all three validation datasets, which demonstrates that our KDIC model is robust for different resolutions image datasets.

**Model Complexity.** Beyond RD performance, we evaluate LIC models' computational efficiency by analyzing parameters, FLOPs, and decoding latency. Table 1 compares the complexity of various LIC models, with FLOPs and latency

| Method | BD-rate [Kodak] | BD-rate [CLIC] | BD-rate [Tecnick] | Params (M) | FLOPs (T) | Dec-Lat. (s) | Throughput (samples/s)↑ |
|---|---|---|---|---|---|---|---|
| MambaVC [49] | -8.11 | -10.94 | -11.82 | 47.88 | 2.10 | 0.425 | 6.55 |
| CCA [19] | -11.99 | -11.40 | -13.53 | 64.89 | 3.28 | 0.385 | 23.28 |
| FTIC [33] | -12.94 | -10.21 | -13.89 | 69.78 | 2.38 | >10 | 23.25 |
| MLIC++ [25] | -11.97 | -12.08 | -15.13 | 116.48 | 2.64 | 0.547 | 27.42 |
| TCM-L [40] | -10.04 | -8.60 | -10.42 | 75.89 | 3.74 | 0.542 | 17.80 |
| KDIC | **-13.71** | **-12.63** | **-16.64** | **47.61** | **1.80** | **0.350** | **36.78** |

Table 1. R-D Performance and Model Complexity of LIC models.

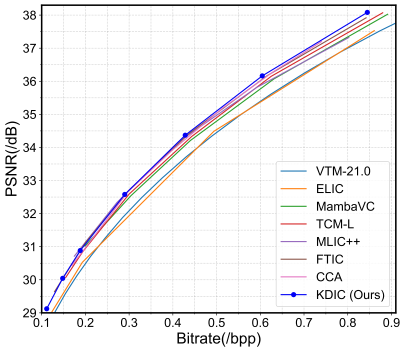


Figure 6. RD curves on the Kodak dataset.

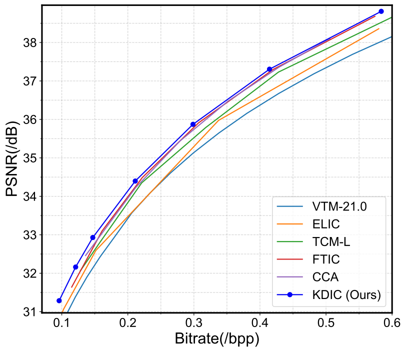


Figure 7. RD curves on the Tecnick dataset.

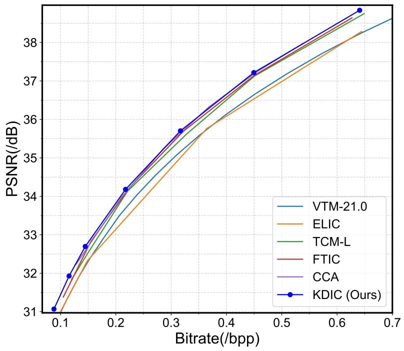


Figure 8. RD curves on the CLIC dataset.

measured using 2K images on an A100 GPU and AMD EPYC 7742 CPU. Training throughput is assessed with a batch size of 8 and 256×256 patches. Our KDIC model surpasses leading LIC models in almost all the metrics. Compared to MLIC++ and TCM-L, KDIC reduces parameters by 59% and 37%, FLOPs by 32% and 52%, and decoding time by 36% and 35%, respectively. These results underscore its practicality for latency-sensitive applications, making KDIC ideal for scenarios demanding high-quality compression and computational efficiency.

|  | KDIC | KDIC (Vanilla) | S2C (Teacher) |
|---|---|---|---|
| BD-rate [Kodak] | -13.71 | -11.21 (↑-2.50) | -14.28 (↓-0.57) |
| BD-rate [CLIC] | -12.63 | -10.05 (↑-2.58) | -12.88 (↓-0.25) |
| BD-rate [Tecnick] | -16.64 | -13.92 (↑-2.72) | -17.20 (↓-0.56) |
| Params (M) | 47.61 | 47.61 | 79.83 (↓ 40%) |
| FLOPs (T) | 1.80 | 1.80 | 4.17 (↓ 57%) |
| Decoding Latency (s) | 0.350 | 0.350 | 0.362 (↓ 3.3%) |
| Throughput (samples/s) | 36.78 | 36.78 | 22.63 (↑ 63%) |

Table 2. The effectiveness of Knowledge Distillation on KDIC.

## 4.3. Effectiveness of Knowledge Distillation

To demonstrate the performance gain of KD, we present detailed comparisons among KDIC, the vanilla-trained version (without distillation), and the teacher model S2CFormer in Tab.2). Compared with the teacher model, KDIC reduces the parameter count by 40% and decreases FLOPs by 57%. However, the BD-rate drops by only 0.57%, 0.25%, and 0.56% across the three datasets, respectively. In contrast, compared with the vanilla-trained version of KDIC, our model achieves BD-rate reductions of more than 2.5% on all datasets. Besides, we also provide RD curves of these three models in Fig. 9-11. Overall, these comparisons demonstrate the effectiveness of our knowledge distillation method.

## 4.4. Ablation Studies

**Distillation Framework.** A series of ablation studies is conducted to evaluate the efficacy of our stage-wise modular distillation framework. We first evaluate three more distillation settings: (1) end-to-end training with the RD loss + MSE feature loss; (2) multi-stage implicit supervision (treating the encoder and decoder as two separate stages); and (3) stage-wise explicit supervision by MSE loss. Tab. 3 demonstrates that our finer-grained stage separation yields superior distillation and explicit MSE supervision does not align well with RD loss in LIC.

| Strategy | Vanilla | (1) | (2) | (3) | SMoDi | Teacher |
|---|---|---|---|---|---|---|
| BD-rate (%) | -11.71 | -11.22 | -12.74 | -12.07 | -13.91 | -14.28 |

Table 3. Distillation framework ablation results.

**Architecture Alignment.** Further ablation experiments were performed to assess the impact of Teacher-guided Student Model Construction. For this study, we selected a completely CNN-based teacher model, S2C-Conv. And we also take KDIC as student model, which is a hybrid structure

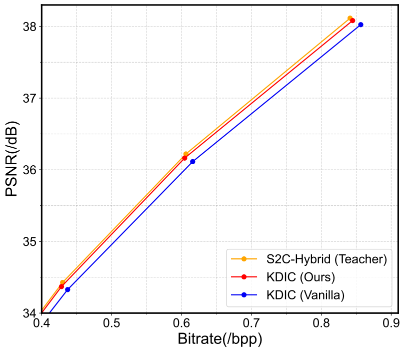


Figure 9. RD curves on the Kodak dataset.

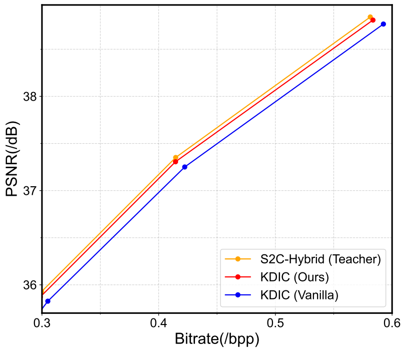


Figure 10. RD curves on the Tecnick dataset.

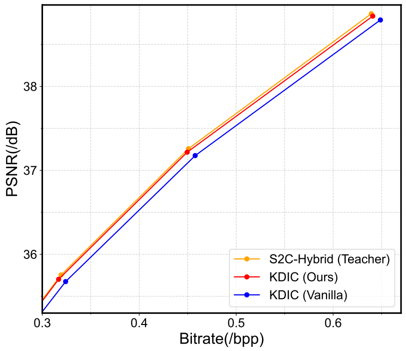


Figure 11. RD curves on the CLIC dataset.

| | Conv-KD | Conv-Vanilla | S2C-Conv (Teacher) | TCM-KD | TCM-Vanilla | TCM-L (Teacher) |
|---|---|---|---|---|---|---|
| BD-rate [Kodak] | -11.49 | -8.77 (↑-2.72) | -12.65 (↓-1.16) | -9.79 | -7.49 (↑-2.30) | -10.04 (↓-0.25) |
| BD-rate [CLIC] | -9.91 | -7.13 (↑-2.78) | -10.98 (↓-1.07) | -8.26 | -6.05 (↑-2.21) | -8.60 (↓-0.34) |
| BD-rate [Tecnick] | -13.45 | -10.25 (↑-3.20) | -14.48 (↓-1.03) | -10.11 | -8.06 (↑-2.05) | -10.42 (↓-0.31) |
| Params (M) | 42.42 | 42.42 | 66.60 (↓ 36%) | 68.14 | 68.14 | 75.89 (↓ 10%) |
| FLOPs (T) | 1.35 | 1.35 | 3.13 (↓ 57%) | 1.84 | 1.84 | 3.74 (↓ 51%) |
| Decoding Latency (s) | 0.331 | 0.331 | 0.356 (↓ 7.0%) | 0.479 | 0.479 | 0.542 (↓ 11.6%) |
| Throughput (samples/s) | 53.85 | 53.85 | 35.49 (↑ 52%) | 23.22 | 23.22 | 17.80 (↑ 30%) |

Table 4. The extension experiments on more structures.

of CNNs and transformers. This absence of architecture alignment resulted in a BD-rate to -11.01%, -10.07% and -13.58 in the three datasets, which was inferior even to the baseline (vanilla) model. This underscores the critical role of architectural alignment in optimizing knowledge transfer.

### 4.5. Extensions on Other Structures

To verify the generalizability of our distillation scheme, we tested it on two other SOTA models: the purely convolution-based S2C-Conv model [11] and the Swin Transformer-based TCM-L [40]. For these two models, we similarly constructed student models by: *1. Reducing the expansion factor in the FFN, 2. Reducing the number of channels in the first and sixth stages, namely $S_1$ and $S_6$,* and *3. Reducing the number of nonlinear transform blocks in the second to fifth stages, namely $N_2$ and $N_3$.* We name these student models Conv-KD and TCM-KD, respectively, and provided their performance and model complexity data in Tab. 4. Additionally, the models that are not trained with knowledge distillation are named Conv-Vanilla and TCM-Vanilla. As shown, our distillation scheme is applicable across multiple models—it greatly reduces FLOPs and model parameters while incurring only a minor RD performance loss compared to the teacher models, and it offers stable improvements over the vanilla-training versions. As we have stated, there can be a lot of KD variants of student models, so we just provide two example KD models in this experiment to demonstrate the extensibility of our KD framework.

## 5. Conclusion

In this work, we addressed the critical challenge of high computational complexity in learned image compression (LIC) models by introducing SMoDi, a novel stage-wise modular distillation framework. By decomposing the compression task into independent sub-tasks and aligning the student model with the teacher model through Teacher-Guided Student Model Construction, we ensured architectural consistency and efficient knowledge transfer. Additionally, our Implicit End-to-end Supervision mechanism enabled adaptive energy compaction and robust bitrate regularization, overcoming the limitations of traditional distillation approaches.

Through the application of SMoDi, we developed KDIC, a lightweight student model derived from the state-of-the-art S2CFormer architecture. Experimental results demonstrated that KDIC achieves top-tier rate-distortion (RD) performance while significantly reducing computational complexity, making it a practical solution for real-world deployment. To the best of our knowledge, this work represents one of the first successful applications of knowledge distillation to LIC, offering a new direction for balancing performance and efficiency in this domain.

**Acknowledgment.** This work is supported by the National Key Research and Development Program of China under Grant 2024YFF0509700, National Natural Science Foundation of China(62471290,62431015,62331014) and the Fundamental Research Funds for the Central Universities.

## References

[1] Sungsoo Ahn, Shell Xu Hu, Andreas Damianou, Neil D Lawrence, and Zhenwen Dai. Variational information distillation for knowledge transfer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9163–9171, 2019. 2, 3

[2] Nicola Asuni, Andrea Giachetti, et al. Testimages: a large-scale archive for testing visual devices and basic image processing algorithms. In *STAG*, pages 63–70, 2014. 6

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

[4] Johannes Ballé, David Minnen, Saurabh Singh, Sung Jin Hwang, and Nick Johnston. Variational image compression with a scale hyperprior. In *International Conference on Learning Representations*, 2018. 1, 2

[5] Jean Bégaint, Fabien Racapé, Simon Feltman, and Akshay Pushparaja. Compressai: a pytorch library and evaluation platform for end-to-end compression research. *arXiv e-prints*, pages arXiv–2011, 2020. 2

[6] Gisle Bjontegaard. Calculation of average psnr differences between rd-curves. In *VCEG-M33*, 2001. 6

[7] 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 Transactions on Circuits and Systems for Video Technology*, 31(10):3736–3764, 2021. 6

[8] A. Browne, Y. Ye, and S. Kim. Algorithm description for versatile video coding and test model 21 (vtm 21), document jvet-af2002. In *Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, 32nd Meeting*, Hannover. 6

[9] Dan Busbridge, Amitis Shidani, Floris Weers, Jason Ramapuram, Etai Littwin, and Russ Webb. Distillation scaling laws, 2025. 2, 5

[10] Fangdong Chen, Yumeng Xu, and Li Wang. Two-stage octave residual network for end-to-end image compression. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 3922–3929, 2022. 2

[11] Liangyu Chen, Xiaojie Chu, Xiangyu Zhang, and Jian Sun. Simple baselines for image restoration. In *European conference on computer vision*, pages 17–33. Springer, 2022. 8

[12] Yunuo Chen, Qian Li, Bing He, Donghui Feng, Ronghua Wu, Qi Wang, Li Song, Guo Lu, and Wenjun Zhang. S2cformer: Reorienting learned image compression from spatial interaction to channel aggregation, 2025. 1, 5, 6

[13] Zhengxue Cheng, Heming Sun, Masaru Takeuchi, and Jiro Katto. Learned image compression with discretized gaussian mixture likelihoods and attention modules. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7939–7948, 2020. 2

[14] CLIC. Workshop and challenge on learned image compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. 6

[15] Zhihao Duan, Ming Lu, Zhan Ma, and Fengqing Zhu. Opening the black box of learned image coders. In *2022 Picture Coding Symposium (PCS)*, pages 73–77. IEEE, 2022. 3

[16] Haisheng Fu, Feng Liang, Jie Liang, Binglin Li, Guohe Zhang, and Jingning Han. Asymmetric learned image compression with multi-scale residual block, importance scaling, and post-quantization filtering. *IEEE Transactions on Circuits and Systems for Video Technology*, 33(8):4309–4321, 2023. 2

[17] Ge Gao, Pei You, Rong Pan, Shunyuan Han, Yuanyuan Zhang, Yuchao Dai, and Hojae Lee. Neural image compression via attentional multi-scale back projection and frequency decomposition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14677–14686, 2021. 2

[18] Qinquan Gao, Yan Zhao, Gen Li, and Tong Tong. Image super-resolution using knowledge distillation. In *Asian Conference on Computer Vision*, pages 527–541. Springer, 2018. 3

[19] Minghao Han, Shiyin Jiang, Shengxi Li, Xin Deng, Mai Xu, Ce Zhu, and Shuhang Gu. Causal context adjustment loss for learned image compression. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. 6, 7

[20] Dailan He, Yaoyan Zheng, Baocheng Sun, Yan Wang, and Hongwei Qin. Checkerboard context model for efficient learned image compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14771–14780, 2021. 2

[21] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5718–5727, 2022. 2, 3, 5, 6

[22] Zibin He, Tao Dai, Jian Lu, Yong Jiang, and Shu-Tao Xia. Fakd: Feature-affinity based knowledge distillation for efficient image super-resolution. In *2020 IEEE International Conference on Image Processing (ICIP)*, pages 518–522. IEEE, 2020. 3

[23] Byeongho Heo, Jeesoo Kim, Sangdoo Yun, Hyojin Park, Nojun Kwak, and Jin Young Choi. A comprehensive overhaul of feature distillation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1921–1930, 2019. 3

[24] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. In *Proceedings of the NIPS 2014 Deep Learning Workshop*, 2014. 2, 3

[25] Wei Jiang and Ronggang Wang. MLIC++: Linear complexity multi-reference entropy modeling for learned image compression. In *ICML 2023 Workshop Neural Compression: From Information Theory to Applications*, 2023. 2, 6, 7

[26] Wei Jiang, Jiayu Yang, Yongqi Zhai, Peirong Ning, Feng Gao, and Ronggang Wang. Mlic: Multi-reference entropy model for learned image compression. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 7618–7627, 2023. 1, 2

[27] Sangwon Jung, Donggyu Lee, Taeeon Park, and Taesup Moon. Fair feature distillation for visual recognition. In *Proceedings*

*of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12115–12124, 2021. 3

[28] Diederik P Kingma. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, 2015. 6

[29] Eastman Kodak. Kodak lossless true color image suite (photocd pcd0992), 1993. Available from http://r0k.us/graphics/kodak/. 6

[30] Akshay Kulkarni, Navid Panchi, Sharath Chandra Raparthy, and Shital Chiddarwar. Data efficient stagewise knowledge distillation. *arXiv preprint arXiv:1911.06786*, 2019. 3

[31] Wonkyung Lee, Junghyup Lee, Dohyung Kim, and Bumsub Ham. Learning with privileged information for efficient image super-resolution. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIV 16*, pages 465–482. Springer, 2020. 3

[32] Chunyi Li, Guo Lu, Donghui Feng, Haoning Wu, Zicheng Zhang, Xiaohong Liu, Guangtao Zhai, Weisi Lin, and Wenjun Zhang. Misc: Ultra-low bitrate image semantic compression driven by large multimodal model, 2024. 2

[33] Han Li, Shaohui Li, Wenrui Dai, Chenglin Li, Junni Zou, and Hongkai Xiong. Frequency-aware transformer for learned image compression. In *The Twelfth International Conference on Learning Representations*, 2024. 2, 6, 7

[34] Han Li, Shaohui Li, Wenrui Dai, Maida Cao, Nuowen Kan, Chenglin Li, Junni Zou, and Hongkai Xiong. On disentangled training for nonlinear transform in learned image compression. In *The Thirteenth International Conference on Learning Representations*, 2025. 3

[35] Jiahao Li, Bin Li, and Yan Lu. Hybrid spatial-temporal entropy modelling for neural video compression. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 1503–1511, 2022. 2

[36] Jiahao Li, Bin Li, and Yan Lu. Neural video compression with diverse contexts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22616–22626, 2023. 2

[37] Shaohui Li, Wenrui Dai, Yimian Fang, Ziyang Zheng, Wen Fei, Hongkai Xiong, and Wei Zhang. Revisiting learned image compression with statistical measurement of latent representations. *IEEE Transactions on Circuits and Systems for Video Technology*, 2023. 3

[38] Dongqin Liu, Wentao Li, Wei Zhou, Zhaoxing Li, Jiao Dai, Jizhong Han, Ruixuan Li, and Songlin Hu. Semantic stagewise learning for knowledge distillation. In *2023 IEEE International Conference on Multimedia and Expo (ICME)*, pages 816–821. IEEE, 2023. 3

[39] Jiaheng Liu, Guo Lu, Zhihao Hu, and Dong Xu. A unified end-to-end framework for efficient deep image compression. *arXiv preprint arXiv:2002.03370*, 2020. 2, 6

[40] Jinming Liu, Heming Sun, and Jiro Katto. Learned image compression with mixed transformer-cnn architectures. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14388–14397, 2023. 2, 6, 7, 8

[41] Yifan Liu, Ke Chen, Chris Liu, Zengchang Qin, Zhenbo Luo, and Jingdong Wang. Structured knowledge distillation for semantic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2604–2613, 2019. 3

[42] Ming Lu, Peiyao Guo, Huiqing Shi, Chuntong Cao, and Zhan Ma. Transformer-based image compression. In *2022 Data Compression Conference (DCC)*, pages 469–469. IEEE, 2022. 2

[43] Haichuan Ma, Dong Liu, Ruiqin Xiong, and Feng Wu. iwave: Cnn-based wavelet-like transform for image compression. *IEEE Transactions on Multimedia*, 22(7):1667–1679, 2019.

[44] Fabian Mentzer, George Toderici, David Minnen, Sergi Caelles, Sung Jin Hwang, Mario Lucic, and Eirikur Agustsson. VCT: A video compression transformer. In *Advances in Neural Information Processing Systems*, 2022. 2

[45] David Minnen and Saurabh Singh. Channel-wise autoregressive entropy models for learned image compression. In *2020 IEEE International Conference on Image Processing (ICIP)*, pages 3339–3343. IEEE, 2020. 1, 2

[46] David Minnen, Johannes Ballé, and George D Toderici. Joint autoregressive and hierarchical priors for learned image compression. *Advances in neural information processing systems*, 31, 2018. 1, 2

[47] Yulei Niu, Long Chen, Chang Zhou, and Hanwang Zhang. Respecting transfer gap in knowledge distillation. In *Advances in Neural Information Processing Systems*, 2022. 2, 5

[48] Yichen Qian, Ming Lin, Xiuyu Sun, Zhiyu Tan, and Rong Jin. Entroformer: A transformer-based entropy model for learned image compression. In *International Conference on Learning Representations*, 2022. 2

[49] Shiyu Qin, Jinpeng Wang, Yimin Zhou, Bin Chen, Tianci Luo, Baoyi An, Tao Dai, Shutao Xia, and Yaowei Wang. Mambavc: Learned visual compression with selective state spaces. *arXiv preprint arXiv:2405.15413*, 2024. 7

[50] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. *International Conference on Learning Representations*, 2015. 2, 3

[51] Yuan Tian, Guo Lu, Yichao Yan, Guangtao Zhai, Li Chen, and Zhiyong Gao. A coding framework and benchmark towards low-bitrate video understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 2

[52] Yuan Tian, Kaiyuan Ji, Rongzhao Zhang, Yankai Jiang, Chunyi Li, Xiaosong Wang, and Guangtao Zhai. Towards all-in-one medical image re-identification. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 30774–30786, 2025.

[53] Yuan Tian, Shuo Wang, and Guangtao Zhai. Medical manifestation-aware de-identification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 26363–26372, 2025. 2

[54] Xiaojie Wang, Rui Zhang, Yu Sun, and Jianzhong Qi. Kdgan: Knowledge distillation with generative adversarial networks. *Advances in neural information processing systems*, 31, 2018. 3

[55] Yueqi Xie, Ka Leong Cheng, and Qifeng Chen. Enhanced invertible encoding for learned image compression. In *Proceedings of the 29th ACM international conference on multimedia*, pages 162–170, 2021. 2

[56] Junho Yim, Donggyu Joo, Jihoon Bae, and Junmo Kim. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4133–4141, 2017. 2, 3

[57] Li Yuan, Francis EH Tay, Guilin Li, Tao Wang, and Jiashi Feng. Revisiting knowledge distillation via label smoothing regularization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3903–3911, 2020. 2, 3

[58] Ali Zafari, Atefeh Khoshkhahtinat, Piyush Mehta, Mohammad Saeed Ebrahimi Saadabadi, Mohammad Akyash, and Nasser M Nasrabadi. Frequency disentangled features in neural image compression. In *2023 IEEE International Conference on Image Processing (ICIP)*, pages 2815–2819. IEEE, 2023. 2

[59] Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. In *International Conference on Learning Representations*, 2017. 3

[60] Linfeng Zhang, Jiebo Song, Anni Gao, Jingwei Chen, Chenglong Bao, and Kaisheng Ma. Be your own teacher: Improve the performance of convolutional neural networks via self distillation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3713–3722, 2019. 3

[61] Linfeng Zhang, Yukang Shi, Zuoqiang Shi, Kaisheng Ma, and Chenglong Bao. Task-oriented feature distillation. *Advances in Neural Information Processing Systems*, 33:14759–14771, 2020. 3

[62] Wenjun Zhang, Guo Lu, Zhiyong Chen, and Geoffrey Ye Li. Generative video communications: Concepts, key technologies, and future research trends. *Engineering*, 2025. 2

[63] Yiwei Zhang, Guo Lu, Yunuo Chen, Shen Wang, Yibo Shi, Jing Wang, and Li Song. Neural rate control for learned video compression. In *The Twelfth International Conference on Learning Representations*, 2023. 1

[64] Yinhao Zhu, Yang Yang, and Taco Cohen. Transformer-based transform coding. In *International Conference on Learning Representations*, 2022. 2

[65] Renjie Zou, Chunfeng Song, and Zhaoxiang 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*, pages 17492–17501, 2022. 2