Neural Image Compression with Text-guided Encoding for both Pixel-level and Perceptual Fidelity

Hagyeong Lee [* 1] Minkyu Kim [* 1] Jun-Hyuk Kim [2] Seungeon Kim [2] Dokwan Oh [2] Jaeho Lee [1]

Abstract

Recent advances in text-guided image compression have shown great potential to enhance the perceptual quality of reconstructed images. These methods, however, tend to have significantly degraded pixel-wise fidelity, limiting their practicality. To fill this gap, we develop a new text-guided image compression algorithm that achieves both high perceptual and pixel-wise fidelity. In particular, we propose a compression framework that leverages text information mainly by text-adaptive encoding and training with joint image-text loss. By doing so, we avoid decoding based on text-guided generative models—known for high generative diversity—and effectively utilize the semantic information of text at a global level. Experimental results on various datasets show that our method can achieve high pixel-level and perceptual quality, with either human- or machine-generated captions. In particular, our method outperforms all baselines in terms of LPIPS, with some room for even more improvements when we use more carefully generated captions.

Project Page: taco-nic.github.io
Code: github.com/effl-lab/TACO

1. Introduction

How can text improve machine vision? This question has been asked repeatedly in various domains of visual computing, giving birth to a number of vision-language models with tremendous multi-modal reasoning and generative capabilities (Radford et al., 2021; Liu et al., 2023a).

The value of text has gained much attention in *image compression* as well, motivated by the astonishing ability of
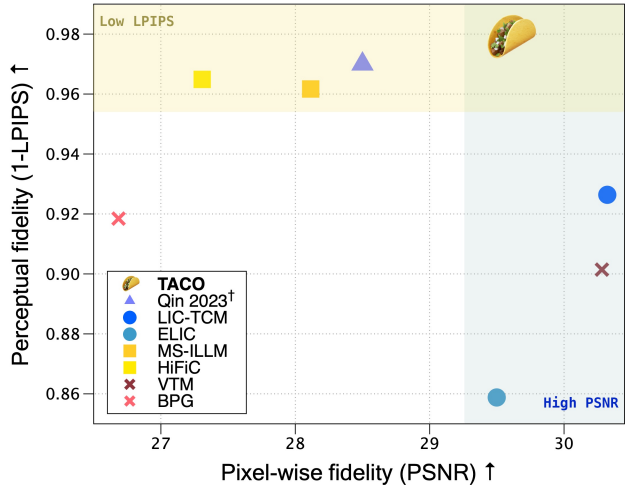

Figure 1. **Pixel-wise fidelity** *vs.* **perceptual fidelity, at 0.40 bpp.** We compare pixel-wise and perceptual fidelity of image compression codecs on MS-COCO 30k. The proposed TACO achieves competitive results in both metrics. The reported figures for Qin 2023 has been measured on MS-COCO 40k, and some figures have been interpolated from nearest bpp models (●: PSNR-focused, ■: perception-focused, ▲: text-guided, ×: handcrafted).

modern text-guided generative models to synthesize realistic images from the given text prompt (Rombach et al., 2022). The conventional way to leverage these models is via text-guided decoding (or *textual transform coding*; Weissman (2023)): We use text-guided generative model as our decoder, and the encoding is done by finding a good textual code. The code may consist of a text prompt plus any additional latent prompt vectors, that can generate the reconstruction that looks close to the original image. As the text can be represented with a very small number of bits, we expect such an approach to be very bit-efficient (Shannon, 1951; Bhown et al., 2018; Weissman, 2023).

Following this intuition, recent works have developed text-guided neural image compression codecs that can reconstruct images with high *perceptual quality*, *i.e.*, either highly realistic (*e.g.*, low FID) or perceptually similar to the original image (*e.g.*, low LPIPS). One line of work shows that using pretrained text-guided generative models as decoders

---

[*]Equal contribution   [1]POSTECH [2]Samsung Advanced Institute of Technology.   Correspondence to: Jaeho Lee <jaeho.lee@postech.ac.kr>.

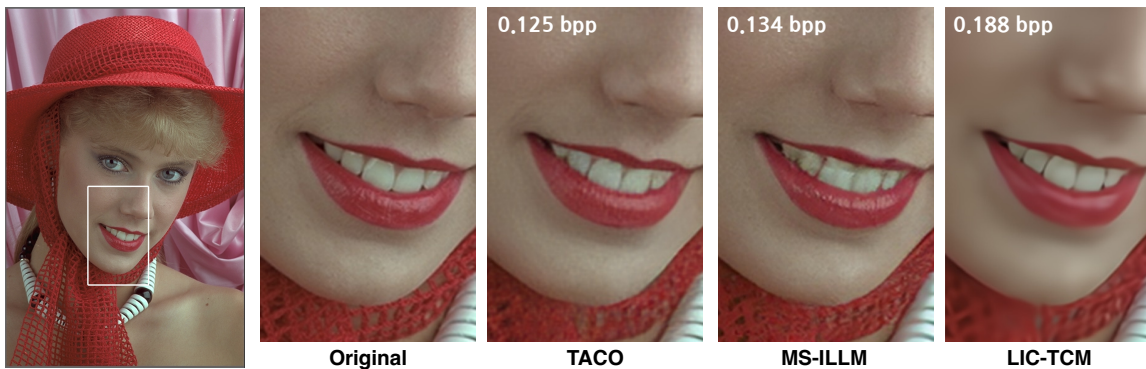*"A woman wearing a red hat and a red dress"*

Figure 2. **Qualitative results.** We compare TACO against MS-ILLM and LIC-TCM, which focuses on perception and PSNR, respectively. TACO uses slightly less bpp than baselines. Comparing with MS-ILLM, TACO tends to suffer less from hallucinated artifacts (see teeth). Comparing with LIC-TCM, TACO can reconstruct sharper details (see lips). See Appendix D for more examples.

can give neural codecs that achieve high realism at extremely low bitrates, *e.g.*, less than 0.01 bits per pixel (Pan et al., 2022; Lei et al., 2023; Careil et al., 2024). Another line of work targets conventional bitrates (over 0.1 bpp) and shows that one can significantly improve the perceptual similarity of reconstructed images to the original image by using text captions as additional side information, where the text information is inserted to the decoder via some plug-in modules (Jiang et al., 2023; Qin et al., 2023).

These approaches, however, commonly suffer from substantial degradations in pixel-wise fidelity, *i.e.*, PSNR. For example, diffusion-based compression codecs tend to achieve 3–5 dB less PSNR than standard neural image compression codecs (Careil et al., 2024). Despite the practical importance of the pixel-wise fidelity (Liu et al., 2023b), it remains unknown whether or how such PSNR degradation can be mitigated for text-guided image compression schemes.

Contribution. In this paper, we fill this gap by developing a text-guided image compression scheme that can achieve high pixel-level and perceptual quality simultaneously (we focus more on the perceptual fidelity, *e.g.*, LPIPS, than on realism). Our key hypothesis is that text-guided decoding *may not* be an effective strategy for PSNR. For high PSNR, we must be able to decode with very small generative diversity, and effectively utilize the semantic information at the global level (*i.e.*, not affecting only local regions). Diffusion-like decoders tend to have too much diversity (Pan et al., 2022), and plug-in approaches tend to have weak control on the global semantics of the reconstruction (Lei et al., 2023).

Thus, we establish an alternative strategy of utilizing text *only for encoding*. Here, the main role of text is to provide additional supervision on *how human perceives the image*; using this information, the encoder can better preserve perceptually meaningful information without discarding it dur-

ing (lossy) compression. By using text only for encoder, we can leverage existing decoder architectures that can generate accurate reconstructions with minimal generative diversity. Furthermore, as we encode the textual information into the code itself, the injected semantic information can affect the whole pixels globally without any extra effort.

Based on this intuition, we propose a simple yet effective encoder-centric text-guided image compression algorithm, coined **TACO** (Text-Adaptive COmpression). TACO transforms a popular PSNR-oriented neural codec architecture (ELIC) into a text-guided one by augmenting the encoder with a text adapter. The text adapter utilizes a pre-trained CLIP encoder (Radford et al., 2021) and bi-directional attention mechanism to inject textual information into the latent code. Then, TACO trains the model from scratch with a joint image-text loss, which encourages the reconstruction to be semantically aligned with the given text.

In our experiments, we find that TACO achieves excellent performance in both pixel-level and perceptual quality at standard bpp levels (*i.e.*, ≥ 0.1). In particular, TACO outperforms all image compression baselines in terms of LPIPS, while achieving competitive performance in pixel-level fidelity and realism when compared to state-of-the-art methods in each metric (Figure 1). This trend holds true for either datasets that come with human-generated image descriptions (*e.g.*, MS-COCO (Lin et al., 2014)), or image-only datasets paired with machine-generated captions (*e.g.*, CLIC (Toderici et al., 2020) captioned by OFA (Wang et al., 2022)). TACO also outperforms conventional LPIPS-focused text-guided baselines (Jiang et al., 2023; Qin et al., 2023).

Our findings suggest that the core value of text in image compression—at least in standard bpp range—may hinge more on its relationship with human perception, than it being an efficient way to store information (Weissman, 2023).

2. Related Work

Measuring Reconstruction Quality. The most popular metric for evaluating the quality of reconstructions is the *pixel-wise distortion* from the original image. Typically, the quantity is measured in mean squared error (MSE) or peak signal-to-noise ratio (PSNR). However, the pixel-wise distortion is often found ill-aligned with the human-perceived image quality (Eskicioglu & Fisher, 1995).

To address this, various *perceptual quality* metrics have been proposed. Roughly, the metrics fall into two categories: *perceptual distortion* and *realism*. Perceptual distortion quantifies how different two images are for human perception. Multi-scale structural similarity (Wang et al., 2003, MS-SSIM) measures the discrepancy of patch-level statistics of both images in various scales. Learned perceptual image patch similarity (Zhang et al., 2018, LPIPS) measures the distortion in the feature space of pretrained neural net classifiers, which are fine-tuned to account for human perception. Similarly, PieAPP trains a model to approximate the human judgement on perceptual similarity (Prashnani et al., 2018).

Realism, on the other hand, attempts to quantify how realistic an image is, without necessarily comparing it with the original image. For example, Fréchet inception distance (Heusel et al., 2017, FID) measures the distance of (Inception) feature distribution of reconstructed images, from that of natural images. Similarly, KID measures the maximum mean discrepancy between the feature distributions (Bińkowski et al., 2018). A recent work by Jayasumana et al. (2023) argues that FID/KID are still misaligned with human perception, and proposes CMMD, an alternative metric that utilizes pre-trained multimodal embeddings.

Neural Image Compression. An early work by Ballé et al. (2017) proposes an autoencoder-like image compression pipeline which achieves competitive pixel-level fidelity and substantially stronger perceptual quality than handcrafted codecs, *e.g.*, JPEG. Subsequent works have developed more advanced hyperpriors on the latent space codes and utilized more advanced autoencoder-like architectures (Ballé et al., 2018; Minnen et al., 2018; Cheng et al., 2020; He et al., 2022; Liu et al., 2023b), which led to even smaller pixel-wise and perceptual distortions on the reproduced image.

Another line of work focuses on developing neural compression codecs for improved realism. Mentzer et al. (2020) propose a GAN-based method to generate highly realistic images at low bitrates. More recent approaches use diffusion model to generate even more realistic images (Hoogeboom et al., 2023; Yang & Mandt, 2023). However, these methods tend to have greater pixel-level or perceptual distortions than autoencoder-based models. MS-ILLM shows that the trade-off between the distortion and realism can be more alleviated by using a more tailored GAN discriminator (Muckley et al.,
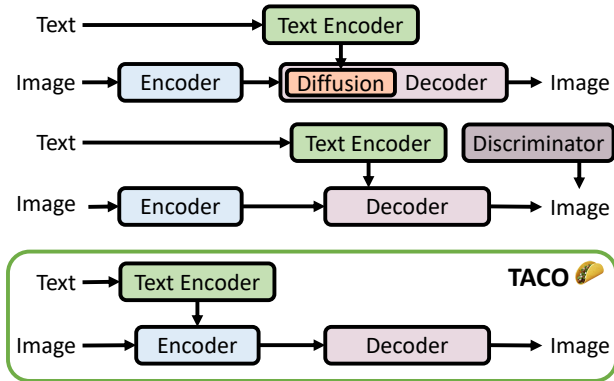

Figure 3. **Text-guided decoding strategies vs. TACO.** (Top) Text-guided decoding with diffusion-based decoders (Careil et al., 2024). (Middle) Text-guided decoder utilizing GAN (Qin et al., 2023). (Bottom) TACO is a much simpler yet effective strategy.

2023). The fundamental limit of this rate-distortion-realism tradeoff has been theoretically studied by Blau & Michaeli (2018), suggesting that both goals may not be achieved simultaneously given a fixed rate.

Text-guided Image Compression. Following the great success of vision-language models, text-guided image compression algorithms have emerged (Bhown et al., 2018; Weissman, 2023). One line of work aims to leverage pre-trained text-guided generative models as decoders. Here, the key is to introduce an appropriate means to control the generation diversity of these models. Pan et al. (2022) utilize the stable diffusion (Rombach et al., 2022), and control the diversity by guiding the reverse step with a JPEG-compressed version of the image. Lei et al. (2023) uses ControlNet as the decoder (Zhang et al., 2023), where the additional control is done by the 'sketch' version of the image, extracted by an edge detector. A concurrent work by Careil et al. (2024) also uses the latent diffusion model, and proposes a more sophisticated local-global encoding for better reconstruction. These models show superior performance in terms of realism (*e.g.*, FID), especially at extremely low bpp.

Another line of work proposes to use decoders that are trained from scratch, using a dataset with image-text pairs such as MS-COCO (Lin et al., 2014). The network architecture typically follows GAN-based image compression codecs (Mentzer et al., 2020), with additional components to insert textual information into the architectures. Jiang et al. (2023) proposes to insert textual information to both encoder and decoder through uni-directional attention-like modules. Qin et al. (2023) insert text only to the decoder through semantic-spatial aware blocks. These works have their main strength on perceptual distortion (*e.g.*, LPIPS) at conventional bpp.

In both lines of work, it has been discovered that one can achieve high realism or perceptual similarity, at the expense of a substantial increase in pixel-level distortion. It remains unanswered whether one can attain similar improvements in the perceptual quality of images from the text, with only minimal degradation in PSNR.

3. Method

We introduce TACO (Text-Adaptive COmpression), a simple yet effective text-guided image compression algorithm that can achieve both high pixel-level and perceptual fidelity.

At a high level, TACO is a simple framework that transforms a conventional PSNR-oriented image compression backbone into a text-guided one. TACO first augments the encoder of the backbone with a text adapter module, which inserts the textual information into the encoder. Then we train the overall network from scratch using a joint image-text loss.

3.1. Background: Autoencoder-based Codec

As the backbone architecture, we use ELIC (He et al., 2022), a PSNR-oriented neural image compression codec. We use ELIC as a base model due to its popularity and performance, and TACO does not rely on any of its structural properties. In principle, we expect TACO to be able to be combined with other base models to enhance their perceptual quality.

Similar to most autoencoder-based codecs, ELIC roughly consists of four components: encoder, quantizer, entropy model, and decoder. An image is processed sequentially by the components as follows. First, the encoder maps an *image x* to a *latent feature y*. The feature is then quantized into $\hat{y}$. The *quantized feature* $\hat{y}$ is (losslessly) compressed into the code and decompressed with the entropy model. Finally, the decoder synthesizes the *reconstruction* $\hat{x}$ from the decompressed feature $\hat{y}$.

$$x \xrightarrow{\text{encode}} y \xrightarrow{\text{quantize \& entropy coding}} \hat{y} \xrightarrow{\text{decode}} \hat{x} \quad (1)$$

The encoder, where TACO injects the text information, is a stack of three residual bottleneck blocks (RBs; He et al. (2016)). The residual blocks are interleaved with five 5×5 convolutional layers (with stride 2) and two attention modules (Cheng et al., 2020). For a more detailed description, see the original paper of He et al. (2022).

3.2. TACO: Text-Adaptive Compression

To enable an effective utilization of the textual information, TACO transforms the backbone image compression model in two ways. First, TACO injects the text information to the encoder of the backbone through *text adapter*. Second, TACO trains the whole model (*i.e.*, backbone model and the text adapter) with the combined loss function of rate,
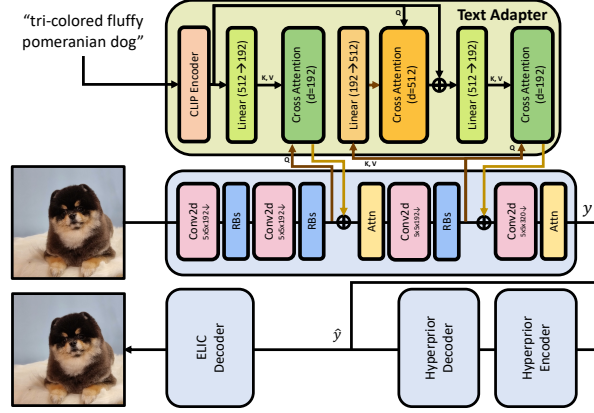

Figure 4. **Text adapter of TACO**. The text adapter first extracts features from the image caption using the CLIP text encoder. The textual features are then injected into the ELIC encoder through multiple cross-attention layers, interleaved with linear layers.

distortion, and the joint image-text loss.

Text Adapter. The text adapter takes in the text segment c associated with the original image x (*e.g.*, image caption) as an input, and injects the text information into the backbone encoder to generate the joint image-text latent feature y.

$$x \xrightarrow{\text{encode}} y \quad \Rightarrow \quad (x, c) \xrightarrow{\text{encode + TACO}} y \quad (2)$$

The text adapter operates in two steps (Figure 4). (1) Embedding: Maps the given text to the joint image-text feature space. (2) Injection: Gradually processes and injects text features into the backbone encoder through multiple layers.

Adapter: Embedding. We leverage a pre-trained CLIP encoder (Radford et al., 2021), which contains rich semantic information on image-text correspondences. By using CLIP features, we expect reductions in the amount of extra computation and data for further reducing the domain gap between features. We use CLIP as frozen without fine-tuning.

More concretely, this step generates a length-m sequence of d-dimensional text tokens. In our experiments, we use $m = 38$, the maximum token length of MS-COCO captions, and $d = 512$, the dimensionality of CLIP encoder outputs.

$$c \xrightarrow{\text{CLIP encoder}} \mathbf{e} := (e_1, \ldots, e_m), \quad e_i \in \mathbb{R}^d. \quad (3)$$

Adapter: Injection. We use a six-layer network consisting of three linear layers and three cross-attention (CA) layers. Here, CAs play a crucial role injecting the text knowledge to images, or vice versa. This use of CA for knowledge injection is inspired by Chen et al. (2023), who uses CA for adapting ViTs to another vision task (dense prediction). In this work, we repurpose CAs for injection across different modalities. Each CA plays the following role:
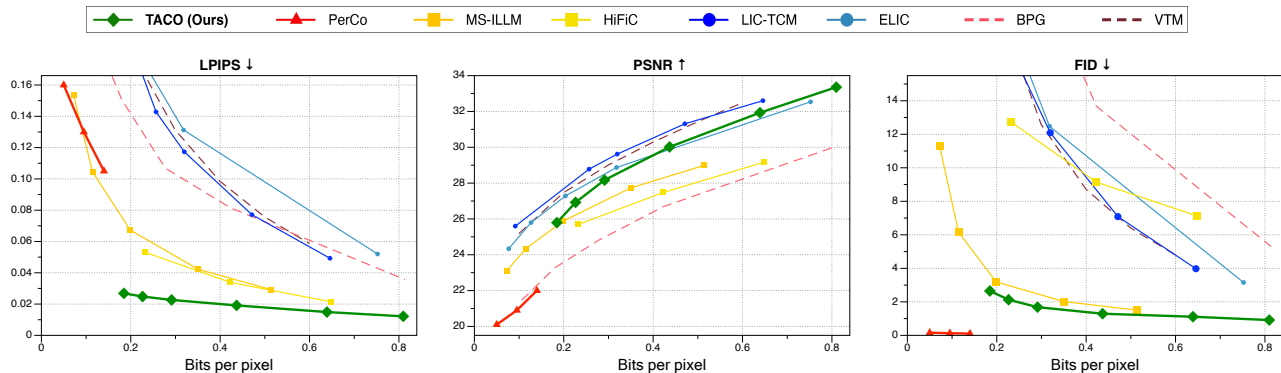
Figure 5. **Compression results: MS-COCO 30k.** TACO achieves the best or competitive results in all metrics. In particular, TACO achieves the best LPIPS among all methods considered, while achieving only ~1dB less PSNR than LIC-TCM. PerCo achieves an impressive FID, but TACO outperforms in terms of both PSNR and LPIPS. We also note a strange under-performance of HiFiC in FID under this setup. This may be due to the resizing operation when measuring FID.

- **CA1: Text→Image.** Updates image features based on text tokens. Given a (h, w, k)-dimensional image feature, we treat each spatial location as an image token. That is, the set of image tokens can be written as:

$$\mathbf{v} = (v_1, \ldots, v_{h \times w}), \quad v_i \in \mathbb{R}^k. \quad (4)$$

CA1 computes query from each $v \in \mathbf{v}$ and keys and values from $e \in \mathbf{e}$. To match the dimensionality of e and v, we put a linear layer before CA1. The output of the layer is

$$\tilde{v}_i = \text{LN}(v_i) + \gamma \odot \text{Attention}(\text{LN}(v_i), \text{LN}(\text{Lin}_1(\mathbf{e}))), \quad (5)$$

where Attention($\cdot$), LN($\cdot$), and Lin($\cdot$) denotes the attention, LayerNorm, and linear layer, respectively. Here, γ is a learnable weight hyperparameter.
- **CA2: Image→Text.** Updates text tokens based on image features which are further processed by the backbone encoder. The operation is similar to CA1, but we compute queries from the text and keys/values from the image.
- **CA3: Text→Image (downsampled).** Same operation as CA1, but with two changes: First, the text tokens have been updated by image features (by CA2). Second, image features have been downsampled, so that text affects more global image features than in CA1.

Joint Image-Text Loss. To train the model, we use a mixture of four different loss functions. Given the original image x, the text description c, the reconstructed image $\hat{x}$, and its quantized latent feature $\hat{y}$, we use the loss

$$r(\hat{y}) + \lambda \cdot d(x, \hat{x}) + k_p \cdot \text{LPIPS}(x, \hat{x}) + k_j \cdot L_j(x, \hat{x}, c) \quad (6)$$

The first three losses are standard: $r(\cdot)$ denotes the rate, $d(\cdot, \cdot)$ denotes the MSE, and LPIPS($\cdot, \cdot$) denotes the LPIPS loss. The fourth loss $L_j(\cdot, \cdot, \cdot)$ is what we call the *joint image-text loss*, which encourages the reconstructed image to be semantically close to the given text and the original image.

Formally, the loss is defined as the following:

$$L_j(x, \hat{x}, c) = L_{\text{con}}(f_I(\hat{x}), f_T(c)) + \beta \cdot \|f_I(x) - f_I(\hat{x})\|_2. \quad (7)$$

Here, the functions $f_I(\cdot)$, $f_T(\cdot)$ denote the CLIP image and text embeddings, and $L_{\text{con}}(\cdot, \cdot)$ denotes the contrastive loss used in CLIP (Radford et al., 2021). In other words, the joint image-text loss is a mixture of (1) the relevance between the text and reconstructed image and (2) the CLIP embedding distance between the original and reconstructed image. We note that Jiang et al. (2023) also uses a loss similar to eq (7), but based on the AttnGAN embedding (Xu et al., 2018).

For better PNSR, we do not use the adversarial loss. Such practice is common in perception-oriented codecs, such as MS-ILLM or HiFiC, but not in PSNR-oriented codecs. TACO works by improving the perceptual quality of PSNR-oriented methods with texts instead of further improving the perceptual quality of already perceptually good models.

4. Experiments

This section is organized as follows. Section 4.1 describes the basic experimental setup. Section 4.2 compares TACO with image compression codecs that do not utilize text. Section 4.3 compares with text-guided codecs; these works do not release code and/or model checkpoints and evaluation setups vary significantly from paper to paper, necessitating a more fine-grained comparison. Finally, Section 4.4 provides various analyses and ablations on TACO.

4.1. Experimental Setup

Datasets. For training, we use the training split of the MS-COCO dataset (Lin et al., 2014), which consists of 82,783 images with five human-annotated captions for each image; we use all five captions for training. We randomly crop each
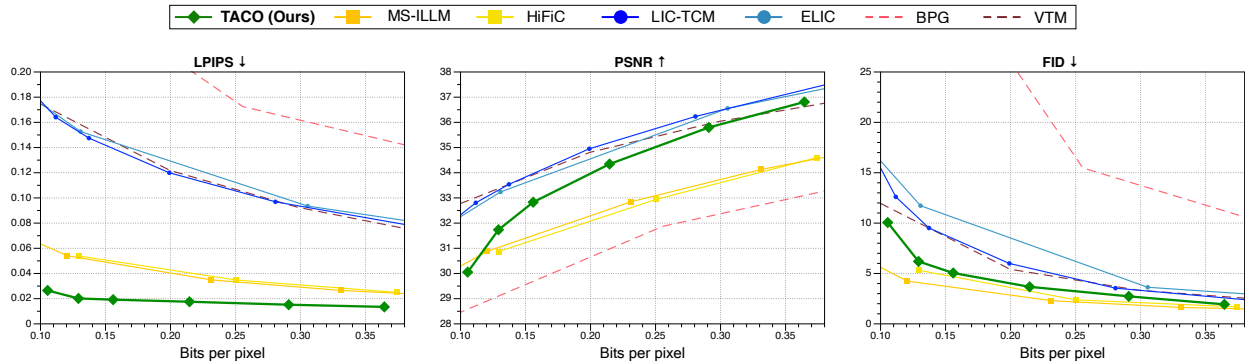
Figure 6. **Compression results: CLIC.** TACO outperforms all other baselines in LPIPS, and achieves close-to-best results in PSNR and FID. In particular, TACO achieves the PSNR very close to LIC-TCM and ELIC, vastly outperforming realism-oriented codecs.
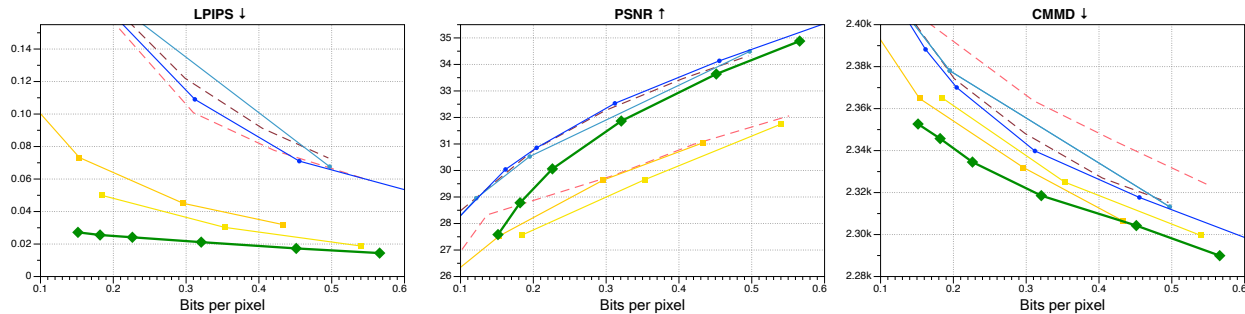

Figure 7. **Compression results: Kodak.** Similar to MS-COCO and CLIC, TACO outperforms baselines with a substantial margin in terms of LPIPs, and closely matches the PSNR of LIC-TCM and ELIC. TACO also performs best in terms of CMMD.

image to 256×256 resolution for training.

For evaluation, we use three different datasets. (1) MS-COCO 30k: We use the "restval" subset of the MS-COCO validation split, which consists of total 30,504 images. For evaluation, we center-crop the images to the 256×256 resolution; we use center drop to more faithfully preserve the semantic information in the image caption.[1] (2) CLIC (Toderici et al., 2020): The test set of the Challenge on Learned Image Compression 2020, consisting of 428 images. We do not crop these images. (3) Kodak dataset (Franzen, 1999): A natural image dataset that consists of 24 images. We also evaluate on these images as-is. We note that CLIC and Kodak do not have any associated captions; we use the image captions generated by OFA (Wang et al., 2022).

Metrics. We focus on three image quality metrics:

- *LPIPS* is a perceptual fidelity metric that measures the distance between the original and reconstructed image in the deep feature space; we use AlexNet features.
- *PSNR* is a pixel-wise fidelity metric, measured in dB.
- *FID* is a realism metric that measures the statistical fi-

delity between two image distributions (original and reconstructed) of inception features. For measuring FID, we follow Muckley et al. (2023) to resize the images to 299×299 resolution before measuring.

As a quantitative realism metric for Kodak dataset, we use a recently proposed CMMD (Jayasumana et al., 2023) instead of FID. This is because FID is known to be unstable in small datasets. For CMMD, we use the publicly available version of CLIP (w.o. projection). We additionally report results on MS-SSIM, PieAPP, and CMMD in the Appendix B.

Optimization & Hyperparameters. We use Adam with batch size 4, and train for 50 epochs. The initial learning rate is set to 10^{-4}, and we decay the learning rate by $1/10$ at epochs 45 and 48. For hyperparameters, we simply use a fixed set $(k_p, k_j, \beta) = (3.5, 0.0025, 40)$ throughout all bpps, instead of conducting an extensive hyperparameter tuning for each bpp. To get models of various bitrates, we train models with $\lambda \in \{4, 8, 16, 40, 90, 150\} \times 10^{-4}$.

4.2. Results: vs. Image Compression Codecs

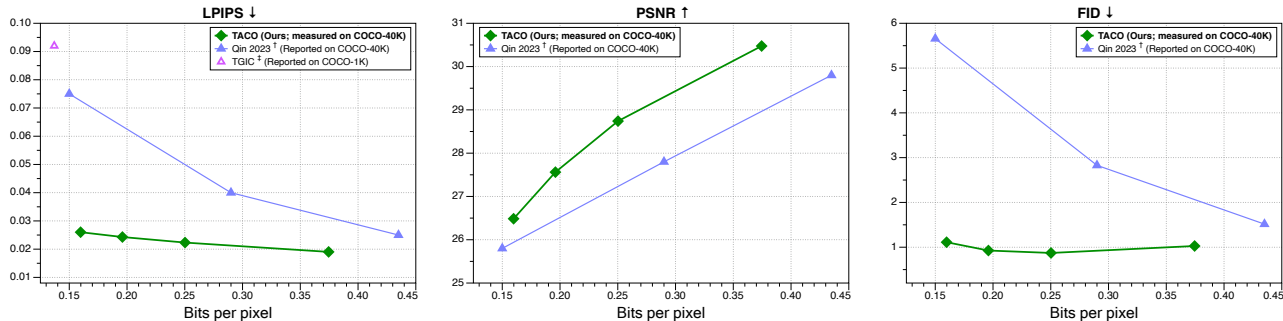We now compare the compression performance of TACO against various image compression codecs, including:

---

[1]This is slightly different from what is called "Dall-E processing" (Ramesh et al., 2021), which have been used by Agustsson et al. (2023), in that they perform random cropping.

Figure 8. **Comparison with text-guided decoding baselines.** We compare with the MS-COCO 40k results reported by Qin et al. (2023), to the performance of TACO on the same dataset. We also compare with the MS-COCO 1k result reported by TGIC (Jiang et al., 2023).
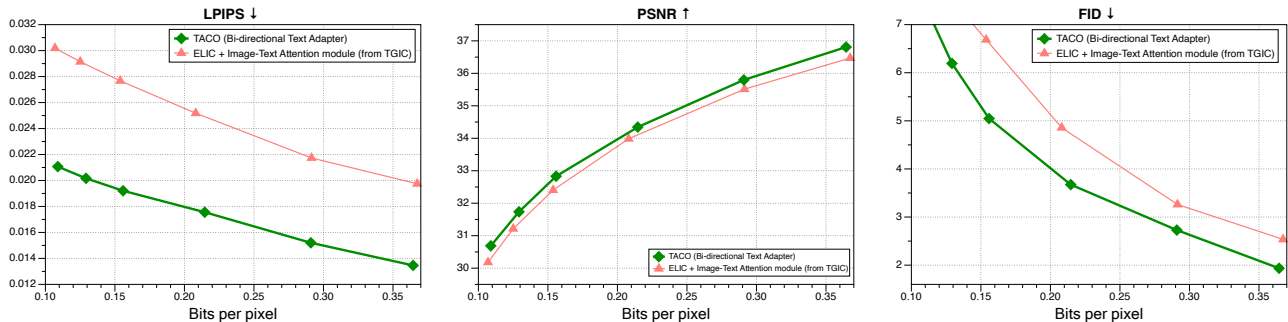

Figure 9. **Text adapter vs. TGIC module.** We compare the effectiveness of the proposed text adapter with the image-text attention module, used by Jiang et al. (2023), on CLIC. Our text adapter performance better in all metrics, when attached on ELIC encoder.

- Handcrafted image compression codecs: VTM, BPG.
- PSNR-oriented neural image compression codecs: ELIC (He et al., 2022), and LIC-TCM (Liu et al., 2023b).
- Perception-oriented neural codecs: HiFiC (Mentzer et al., 2020) and MS-ILLM (Muckley et al., 2023).

We have used the official checkpoints for evaluation, except for ELIC (for which no official checkpoint is available).[2] We provide quantitative comparisons on MS-COCO 30k, CLIC, and Kodak on Figures 5 to 7, respectively.

We observe that, in terms of LPIPS (perceptual fidelity), TACO outperforms all baselines in all datasets. For example, on MS-COCO 30k, TACO achieves 0.025 LPIPS at 0.226 bpp, which is less than half achieved by the best baseline (HiFiC), achieving 0.053 at 0.233 bpp. In terms of PSNR, TACO closely achieves the performance of ELIC and LIC-TCM, falling within the 1dB range from these baselines. In realism, TACO outperforms baselines in MS-COCO and Kodak, but not in CLIC; the gap In CLIC, however, tends to be very small, especially at a high bpp regime.

---

[2]We use the checkpoints from the following Github repo: https://github.com/VincentChandelier/ELiC-ReImplemetation

4.3. Results: vs. Text-guided Decoding

We now compare the performance of TACO with text-guided codecs. In particular, we compare with codecs whose goal is to achieve high LPIPS at non-extreme bpp (*i.e.*, over 0.1 bpp), such as TGIC (Jiang et al., 2023) and "Qin 2023" (Qin et al., 2023). Two things make a direct comparison difficult: (1) The baselines do not release official code or have model checkpoints publicly available. (2) The baselines report performance on non-unified setups: TGIC reports performance on MS-COCO 1k (along with CUB and Oxford-102), consisting of 1k randomly drawn test set samples. Qin et al. (2023) reports performance on MS-COCO 40k, *i.e.*, the whole validation set, without any cropping.

We focus on comparing with (more reproducible) Qin 2023 baseline. We evaluate TACO on MS-COCO 40k and compare it with the results reported in Qin et al. (2023). From Figure 8, we observe that TACO achieves much better performance than the previous text-guided baseline, based on the text-guided decoding strategy. This is somewhat surprising, as the original intention of using text-guided encoding has been on preventing the degradations in PSNR; instead, we also achieve better LPIPS and FID. This improvement can be attributed to the effectiveness of the text-adapter-based
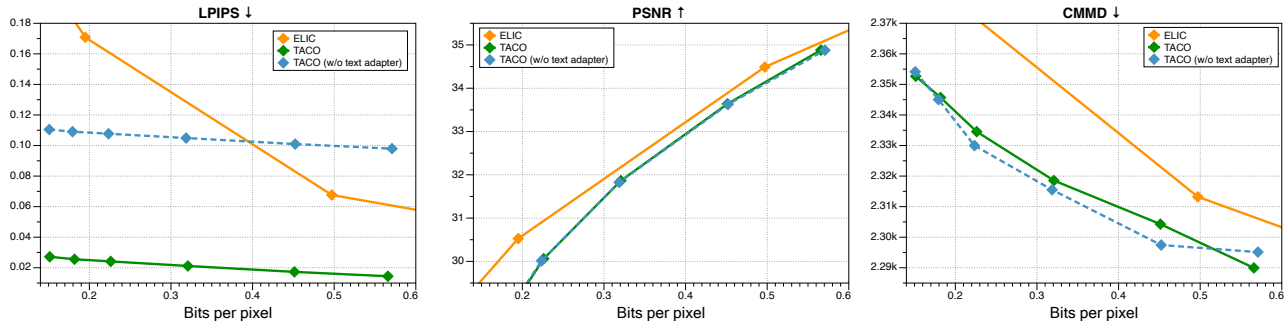
Figure 10. **Without text adapter.** We observe both CMMD and LPIPS substantially degrade, while there is a tiny gain in PSNR.
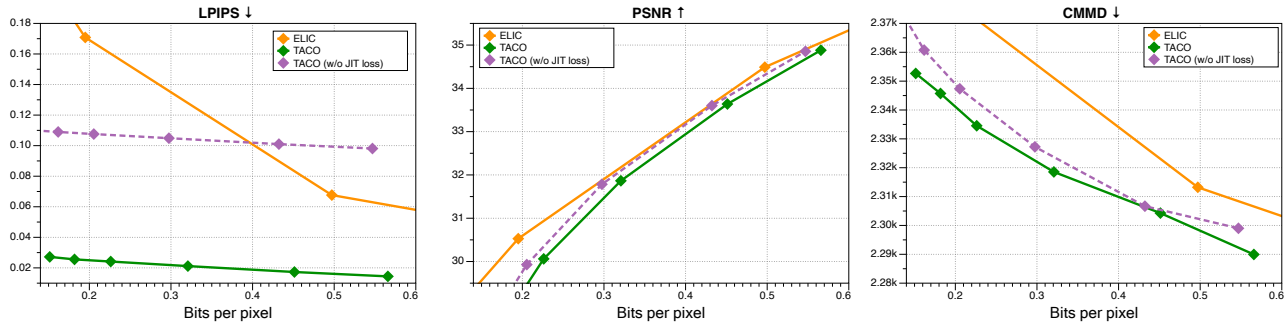

Figure 11. **Without joint image text loss.** We observe LPIPS severely degrades, while PSNR remains similar, with a tiny gain in CMMD.

knowledge injection strategy.

4.4. Ablation Studies

To validate that both our proposed text adapter and joint image-text loss improve the performance, we conduct an ablation study. Through experiments on the Kodak dataset, we confirm that both components play essential roles.

Without text adapter. From Figure 10, we observe that perceptual fidelity greatly decreases without a text adapter (more than 4× in LPIPS). This gap is well-aligned with our intuition that the textual information plays an important role in how human perceives visual signals.

Without joint image-text loss. From Figure 11, we observe that the joint image-text loss contributes in improving not only the perceptual fidelity, but also realism (with a tiny degradation in PSNR). We hypothesize that the manifold-level information transferred from the CLIP embedding, which is trained with a large-scale dataset, helps the reconstructed image to remain realistic.

4.5. Further Analyses

We now ask ourselves several questions regarding the validity and practicality of the proposed method, TACO.

Q1. Is the adapter architecture (Section 3.2) effective?
A. Yes. It outperforms the baseline method for text injection. We compare it to another text injection mechanism introduced in TGIC (Jiang et al., 2023), which utilizes several convolutional layers to inject text information into the image encoder (coined "image-text attention module"). We compare the performance of these two modules for injecting text information to the encoder, in Figure 9. From the figure, we observe that the proposed text adapter is indeed much more effective than what TGIC uses.

Q2. Does TACO preserve textual information better?
A. Yes, and No. TACO-compressed images tend to preserve textual information much better than PSNR-focused methods, similar to LIC-TCM. We measure various text similarity scores between machine-generated captions for the original and reconstructed image (Table 1); we use OFA for this purpose. We find that the images compressed by TACO preserve text semantics better than LIC-TCM and ELIC. On the other hand, the comparison with MS-ILLM is inconclusive; TACO performs slightly worse, but with slightly less bpp. In general, we find that models with high perceptual quality tend to preserve textual information better than PSNR-oriented compression codecs.

Q3. How much computation does TACO add?
A. ∼ 5–10%, depending on the image resolution. We

Table 1. **Comparing text similarity.** We compare the text similarity scores for the CLIC images compressed by TACO and other baseline codecs. We find that TACO achieves much higher text similarity score than LIC-TCM, but is slightly worse than MS-ILLM that uses slightly more bits per pixels.

| | BLEU-4 (↑) | CIDEr-D (↑) | SPICE (↑) |
	(Papineni et al., 2002)	(Vedantam et al., 2015)	(Anderson et al., 2016)
LIC-TCM (0.2810)	0.76	7.44	0.80
MS-ILLM (0.2314)	0.85	8.39	0.87
ELIC (0.3055)	0.78	7.58	0.81
TACO (0.2146)	0.84	8.21	0.86

Table 2. **Computational cost of TACO.** We compare the wall-clock encoding and decoding latency of TACO with baseline image compression codecs. We average over 100 MS-COCO images (256×256), randomly drawn from the validation split (0.0148 bpp). We also compare the computational cost when encoding a high-resolution image ('High' in table), observing that the increasing cost of attention remains relatively small (+10.2% → +4.8%).

	Enc. (ms)	Enc.@High (ms)	Dec. (ms)	Total (ms)
LIC-TCM	112.07	960.99	125.26	237.33
MS-ILLM	70.39	800.45	53.74	124.14
ELIC	71.35	793.41	102.07	173.42
TACO	78.60 (+10.2%)	832.07 (+4.8%)	102.98	181.58

compare the wall-clock encoding and decoding speed of the proposed TACO with the baselines in Table 2. We have measured the latency on a Linux GPU server equipped with one NVIDIA GeForce RTX 3090 and AMD EPYC 7313 16-Core CPU. From the table, we find that TACO indeed introduces some latency in compression speed. In particular, we observe that the encoding time increases by 7ms over the vanilla ELIC. We note that the overhead, however, is not very big. TACO still runs much faster than LIC-TCM and remains comparable with MS-ILLM. Most of the overhead comes from the CLIP embedding step, which can be effectively parallelized with image encoding steps. Furthermore, we have used the smallest version of CLIP, which can be run fast on conventional GPU devices. We also compare the encoding speed when using a high-resolution image (1788×2641, second column in the Table 2). As a result, the computational cost of attention remains relatively small and we find that the computational cost of CLIP remains constant with respect to the resolution.

Q4. How many parameters are used in TACO?
A. Not too big—smaller than perception-oriented codecs.
We compare the number of models in Table 3. TACO adds 64.82M parameters to the ELIC, where 63.17M comes from the CLIP and 1.65M comes from the text adapter. In total, TACO has 101.75M parameters, which requires ~400MBs to be loaded on memory. While this is larger than vanilla ELIC, it is ~ 80% smaller than other perception-oriented codecs, such as MS-ILLM, and HiFiC.

Table 3. **Memory efficiency of TACO.** We compare the model parameter sizes of the TACO and baselines. TACO shows superior memory efficiency than perception-oriented codecs.

Modules	Parameters (M)
ELIC	36.93
LIC-TCM	45.41
HiFiC/MS-ILLM	181.72
TACO	101.75

Table 4. **Caption dependency.** We compare TACO reconstruction results with four different captions generation methods, spanning from human to GPT-4. Human and GPT-4 achieves the best results when measured on 100 random samples of MS-COCO, but other captioning methods work very competitively (0.0148 bpp). Underlined denotes the best result.

| | Human | OFA | BLIP-2 | GPT-4 |
	(Chen et al., 2015)	(Wang et al., 2022)	(Li et al., 2023)	(Achiam et al., 2023)
LPIPS	0.0435	0.0435	0.0435	0.0435
PSNR	27.42	27.41	27.41	27.42

Q5. Can we use other captioning models?
A. Yes. In Table 4, we compare the compression quality of TACO utilizing captions generated by various methods. We observe that GPT-4 achieves a human-like result, but other captioning models perform very competitively. From this observation, we conjecture that the compression quality of TACO is less dependent on its writing style, but more on the core content of the sentences.

5. Conclusion

In this paper, we have studied how one can utilize auxiliary text information when compressing images. We find that, in order to achieve high PSNR as well as high perceptual quality, utilizing the text to generate better codes is a simple yet effective strategy; using text to guide the decoding procedure may be much more challenging task.

Limitations and future work. There are several limitations of the proposed framework, which we hope to address in the future work. First, the additional encoding cost for handling the text scales quadratically with respect to the sequence length. This is due to the design of our text injection mechanism which computes cross-attention between image and textual features. In the cases where we expect the text sequence to be very long, *e.g.*, video-like data with voice transcripts as text, the encoding time can get prohibitively long. Another limitation is the scalability of training. Currently, we train TACO using the image-text dataset, which is substantially scarcer than image-only or text-only dataset. To overcome this problem, using a captioning model to generate image-text pairs for training may be a good baseline.

Acknowledgements

Impact Statement

Our paper aims to advance the general field of machine learning and data compression. We thus expect many societal consequences to follow. However, we do not feel any specific consequence should be highlighted further.

References

Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al. GPT-4 technical report. *arXiv preprint 2303.08774*, 2023.

Agustsson, E., Minnen, D., Toderici, G., and Mentzer, F. Multi-Realism Image Compression with a Conditional Generator. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

Anderson, P., Fernando, B., Johnson, M., and Gould, S. SPICE: Semantic propositional image caption evaluation. In *European Conference on Computer Vision*, 2016.

Ballé, J., Laparra, V., and Simoncelli, E. P. End-to-end optimized image compression. In *International Conference on Learning Representations*, 2017.

Ballé, J., Minnen, D., Singh, S., Hwang, S. J., and Johnston, N. Variational image compression with a scale hyperprior. In *International Conference on Learning Representations*, 2018.

Bhown, A., Mukherjee, S., Yang, S., Chandak, S., Fischer-Hwang, I., Tatwawadi, K., Fan, J., and Weissman, T. Towards improved lossy image compression: Human image reconstruction with public-domain images. *arXiv preprint 1810.11137*, 2018.

Bińkowski, M., Sutherland, D. J., Arbel, M., and Gretton, A. Demystifying MMD GANs. *International Conference on Learning Representations*, 2018.

Blau, Y. and Michaeli, T. The perception-distortion tradeoff. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018.

Careil, M., Muckly, M. J., Verbeek, J., and Lathuilière, S. Towards image compression with perfect realism at ultra-low bitrates. In *International Conference on Learning Representations*, 2024.

Chen, X., Fang, H., Lin, T.-Y., Vedantam, R., Gupta, S., Dollár, P., and Zitnick, C. L. Microsoft COCO captions: Data collection and evaluation server. In *arXiv preprint 1504.00325*, 2015.

Chen, Z., Duan, Y., Wang, W., He, J., Lu, T., Dai, J., and Qiao, Y. Vision transformer adapter for dense predictions. In *International Conference on Learning Representations*, 2023.

Cheng, Z., Sun, H., Takeuchi, M., and Katto, J. Learned Image Compression with Discretized Gaussian Mixture Likelihoods and Attention Modules. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

Eskicioglu, A. M. and Fisher, P. S. Image quality measures and their performance. *IEEE Transactions on Communications*, 1995.

Franzen, R. Kodak lossless true color image suite, 1999.

He, D., Yang, Z., Peng, W., Ma, R., Qin, H., and Wang, Y. ELIC: Efficient learned image compression with unevenly grouped space-channel contextual adaptive coding. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2016.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. *Advances in Neural Information Processing Systems*, 2017.

Hoogeboom, E., Agustsson, E., Mentzer, F., Versari, L., Toderici, G., and Theis, L. High-fidelity image compression with score-based generative models. *arXiv preprint 2305.18231*, 2023.

Jayasumana, S., Ramalingam, S., Veit, A., Glasner, D., Chakrabarti, A., and Kumar, S. Rethinking FID: Towards a better evaluation metric for image generation. *arXiv preprint 2401.09603*, 2023.

Jiang, X., Tan, W., Tan, T., Yan, B., and Shen, L. Multi-modality deep network for extreme learned image compression. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2023.

Lei, E., Uslu, Y. B., Hassani, H., and Bidokhti, S. S. Text + Sketch: Image compression at ultra low rates. *arXiv preprint 2307.01944v1*, 2023.

Li, J., Li, D., Savarese, S., and Hoi, S. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *Proceedings of the International Conference on Machine Learning*, 2023.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. Microsoft COCO: Common objects in context. In *European Conference on Computer Vision*, 2014.

Liu, H., Li, C., Wu, Q., and Lee, Y. J. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, 2023a.

Liu, J., Sun, H., and Katto, J. Learned image compression with mixed transformer-cnn architectures. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023b.

Mentzer, F., Toderici, G. D., Tschannen, M., and Agustsson, E. High-fidelity generative image compression. In *Advances in Neural Information Processing Systems*, 2020.

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

Muckley, M. J., El-Nouby, A., Ullrich, K., Jégou, H., and Verbeek, J. Improving statistical fidelity for neural image compression with implicit local likelihood models. In *Proceedings of the International Conference on Machine Learning*, 2023.

Pan, Z., Zhou, X., and Tian, H. Extreme generative image compression by learning text embedding from diffusion models. *arXiv preprint 2211.07793v1*, 2022.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2002.

Prashnani, E., Cai, H., Mostofi, Y., and Sen, P. PieAPP: Perceptual image-error assessment through pairwise preference. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018.

Qin, S., Chen, B., Huang, Y., An, B., Dai, T., and Via, S.-T. Perceptual image compression with cooperative cross-modal side information. *arXiv preprint arXiv:2311.13847*, 2023.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. Learning transferable visual models from natural language supervision. In *Proceedings of the International Conference on Machine Learning*, 2021.

Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., and Sutskever, I. Zero-shot text-to-image generation. In *Proceedings of the International Conference on Machine Learning*, 2021.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. High-resolution image synthesis with latent diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

Shannon, C. E. Prediction and entropy of printed English. *Bell System Technical Journal*, 1951.

Toderici, G., Theis, L., Johnston, N., Agustsson, E., Mentzer, F., Ballé, J., Shi, W., and Timofte, R. CLIC 2020: Challenge on learned image compression, 2020, 2020.

Vedantam, R., Zitnick, C. L., and Parikh, D. CIDEr: Consensus-based image description evaluation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2015.

Wang, P., Yang, A., Men, R., Lin, J., Bai, S., Li, Z., Ma, J., Zhou, C., Zhou, J., and Yang, H. OFA: Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. In *Proceedings of the International Conference on Machine Learning*, 2022.

Wang, Z., Simoncelli, E. P., and Bovik, A. C. Multiscale structural similarity for image quality assessment. In *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*. IEEE, 2003.

Weissman, T. Toward textual transform coding. *arXiv preprint 2305.01857v1*, 2023.

Xu, T., Zhang, P., Huang, Q., Zhang, H., Gan, Z., Huang, X., and He, X. AttnGAN: Fine-grained text to image generation with attentional generative adversarial networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018.

Yang, R. and Mandt, S. Lossy image compression with conditional diffusion models. *Advances in Neural Information Processing Systems*, 2023.

Zhang, L., Rao, A., and Agrawala, M. Adding conditional control to text-to-image diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

Zhang, R., Isola, P., Efros, A. A., Shechtman, E., and Wang, O. The unreasonable effectiveness of deep features as a perceptual metric. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018.

Appendix

A. Additional Experimental Details

Implementation of TACO. As illustrated in Figure 4, the base model we use is a version of ELIC, which has an intermediate feature dimension of 192 and a final feature map dimension of 320. The CLIP encoder we use is the "CLIPTextModel" with a version of 'openai/clip-vit-base-patch32,' available through the HuggingFace. We have set the token length to be 38, therefore getting the text embedding sequences with the size of $\mathbb{R}^{38 \times 512}$ when inputting the caption to the CLIPTextModel.

Baselines. As mentioned in the main text, for neural codecs, we mostly use the official model checkpoints. For BPG, we have used version 0.9.8. For VTM, we have used version 19.2.

B. Additional Metrics on Compression Results

We additionally report more compression results, MS-SSIM, PieAPP, and CMMD scores, measured on the MS-COCO 30k, CLIC, and Kodak datasets.
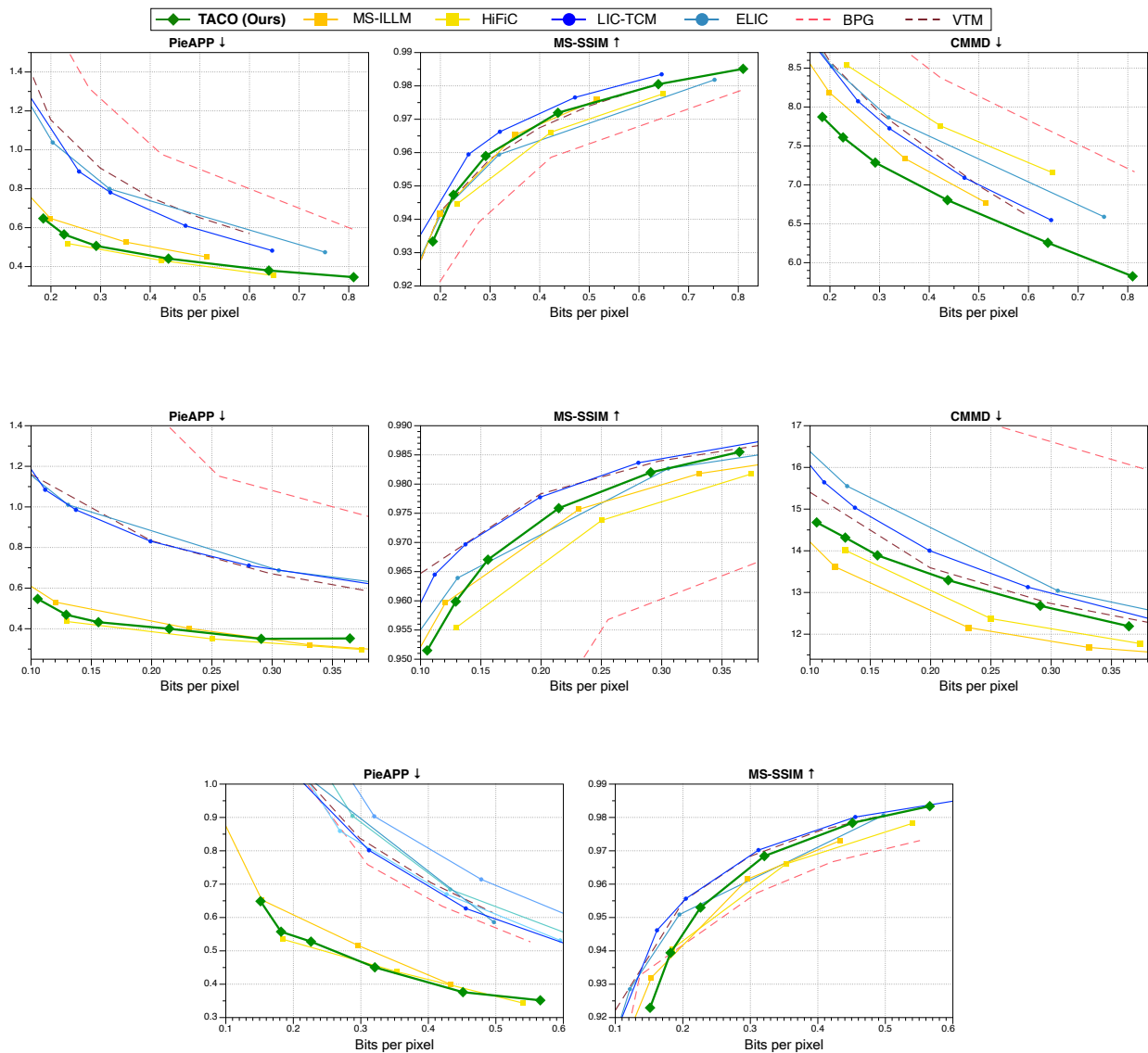

Figure A1. **Additional quality metrics result on: MS-COCO 30k (*top*), CLIC (*middle*), Kodak (*bottom*)**

C. Additional Ablation Study

Freezing base model. We test whether freezing the base model (ELIC) to be a fully image-trained model is beneficial for performance or not. We find that this is not true (see Figure A2).
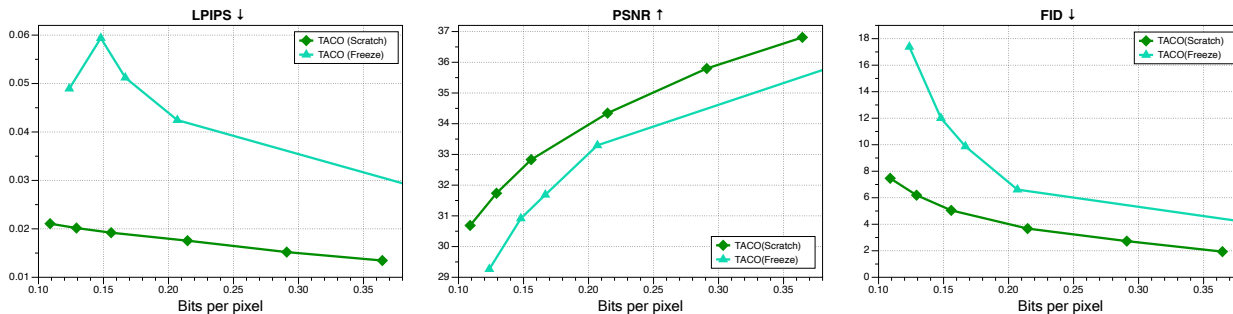

Figure A2. Ablation result: freezing base model and only training text adapter
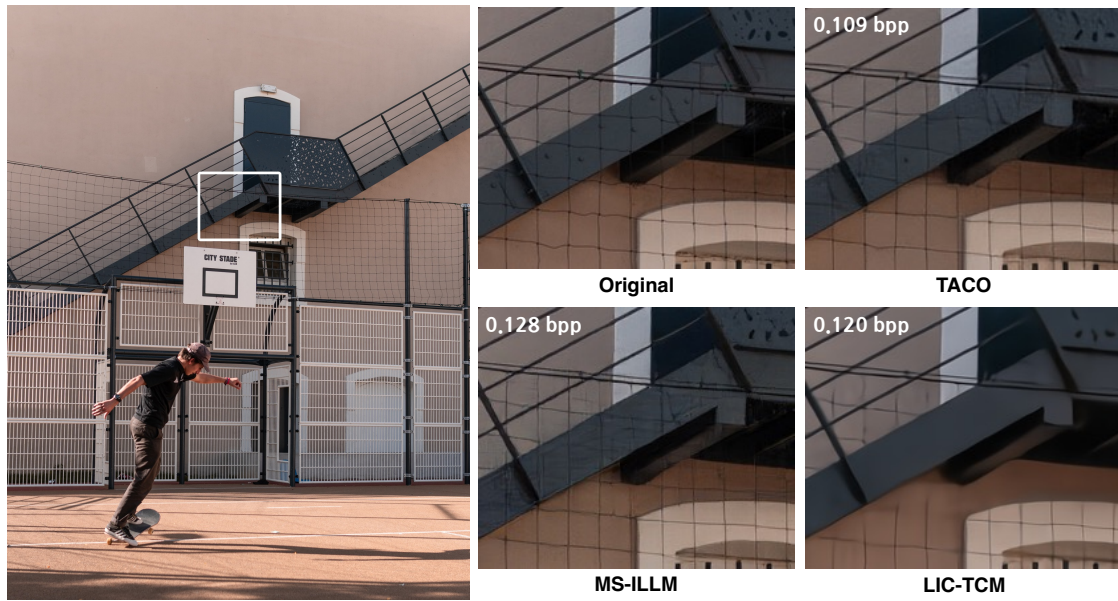
D. Additional Qualitative Results

We provide additional visualizations of the compression results of TACO and baselines in Figure A4. As has been observed in Figure 2, we confirm that TACO suffers from fewer compression artifacts than MS-ILLM while reconstructing sharper details than LIC-TCM. We find that TACO consistently provides high-quality reconstructions, while LIC-TCM tends to overly smooth out the textures (see, e.g., the white wall under the cat) and MS-ILLM hallucinates small details (see, e.g., eyes of the cat).

- In the top display, we draw attention to two features: "the reflections on glasses," and "the texture of eyebrow and hair." MS-ILLM tends to hallucinate flat reflections on the glasses, and LIC-TCM tends to generate blurry eyebrow and hair.
- In the middle display, we focus on the black metal staircase behind the net. MS-ILLM generates wavy textures on the metal, and LIC-TCM removes the net in front of the staircase. TACO also generates some hallucinative patterns but to a lesser degree. Also, we note that TACO is the only compression method that reconstructs (any) stud on the staircase.
- In the bottom display, we highlight the gold hinge. MS-ILLM smooths out the details on the hinge, and LIC-TCM generates a blurry image overall.

"A young woman sitting on the grass wearing a hat and glasses"

0.074 bpp 0.094 bpp 0.098 bpp

Original TACO MS-ILLM LIC-TCM

"A young man riding a skateboard in a parking lot"

0.109 bpp 0.128 bpp 0.120 bpp

Original TACO MS-ILLM LIC-TCM

"An old piano with books and a lamp on it"

0.065 bpp 0.080 bpp 0.077 bpp

Original TACO MS-ILLM LIC-TCM

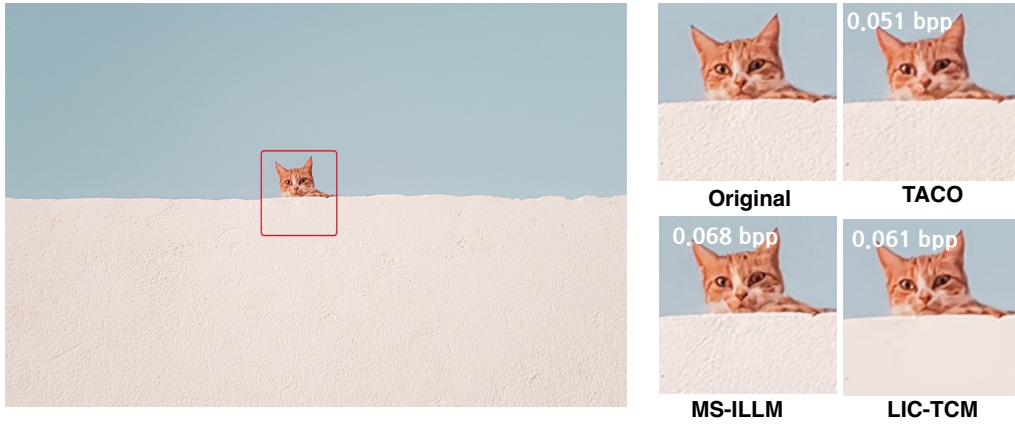*Figure A3.* **Additional Qualitative Results.**

"A large body of water with palm trees on an island"

"A cat sitting on top of a white sand field"
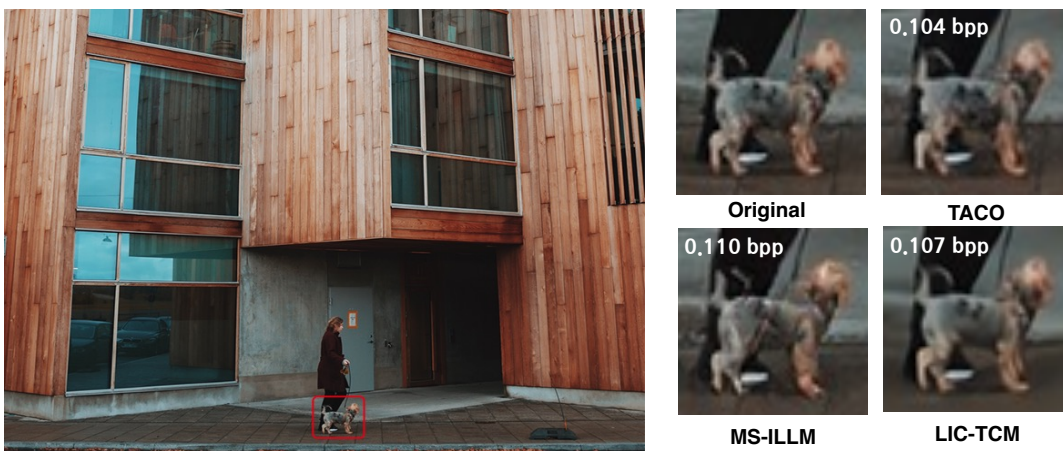

"A person walking a dog in front of a building"

Figure A4. **Additional Qualitative Results.**