DLF: Extreme Image Compression with Dual-generative Latent Fusion Supplementary Material

This document provides the supplementary material for the proposed Dual-generative Latent Fusion (DLF) model, which introduces a novel dual-branch coding approach for extreme image compression. It provides comprehensive details on the training strategy, model architecture, experiments, and additional evaluations.

A. Training Details

This section outlines the detailed training strategy. We train our model using the Open Images v4 training dataset [8]. Due to the storage limitations, we randomly sample 400,000 images from this dataset for training.

A.1. Stage 1: Latent Alignment

To ensure that the detail branch captures the most significant contents while discarding minor ones, we impose a rate constraint on the detail features. Consequently, we design the latent domain rate-distortion loss as follows:

$$\mathcal{L}_{\text{Stage-1}} = ||\hat{h} - \tilde{h}||_2^2 + \lambda \cdot \mathcal{R}(\hat{y}_d) \tag{1}$$

where $\hat{h}$ is the decoded latent generated by the latent adaptor, and the supervision latent $\tilde{h}$ is obtained from a pretrained auxiliary encoder (i.e., the VQGAN encoder [4]). Additionally, $\mathcal{R}(\hat{y}_d)$ denotes the bitrate of the quantized detail latent $\hat{y}_d$, estimated using the quadtree-partition-based spatial context module [10]. The weight λ controls the trade-off between rate and distortion items.

At the start of this stage, we load the pretrained weights of the 1-D tokenizer [21] into the semantic branch. These weights are fixed during this stage to maintain their initialization benefits. This approach provides a strong initialization, accelerating the training process and aiding faster convergence. The semantic codebook size is set to 4096.

To prevent the detail branch from discarding excessive information at the training beginning, we employ a multi-stage λ strategy. Specifically, we initiate the training with a λ value of 0.001 for the first 10,000 steps. Subsequently, the λ value increases gradually from 2.0 to 24.0 over 90,000 steps. Finally, we maintain $\lambda = 24.0$ for the remaining 400,000 steps. We apply this strategy to all rate points during this stage, serving as an initialization for the subsequent stage. During this stage, we use randomly cropped 256×256 images with a batch size of 16, setting the learning rate to 4.0×10^{-5} with the Adam optimizer [7].

A.2. Stage 2: End-to-end Fine-tune

In this stage, we fine-tune the entire model, including all parameters in both branches, the latent adaptor, and the pixel generator. To achieve superior reconstruction quality, we employ the pixel-domain rate-distortion loss:

$$\mathcal{L}_{\text{Stage-2}} = \mathcal{L}_{\text{pixel}} + \mathcal{L}_{\text{codebook}} + \lambda \cdot \mathcal{R}(\hat{y}_d) \tag{2}$$

Here, the pixel-domain distortion loss $\mathcal{L}_{\text{pixel}}$ and the codebook loss $\mathcal{L}_{\text{codebook}}$ in the semantic branch are defined as:

$$\mathcal{L}_{\text{pixel}} = ||x - \hat{x}|| + \mathcal{L}_{\text{LPIPS}}(x, \hat{x}) + \lambda_{\text{adv}} \cdot \mathcal{L}_{\text{adv}}(x, \hat{x}) \tag{3}$$

$$\mathcal{L}_{\text{codebook}} = ||\text{sg}(y_s) - \hat{y_s}|| + \beta \cdot ||\text{sg}(\hat{y_s}) - y_s|| \tag{4}$$

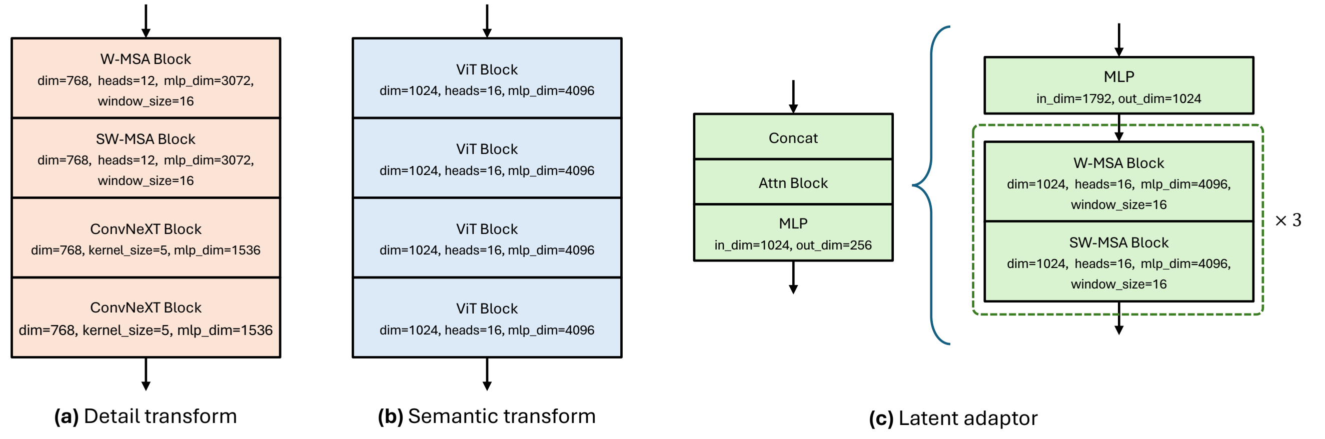
The term $\mathcal{L}_{\text{adv}}$ corresponds to the adaptive adversarial loss [4], weighted by $\lambda_{\text{adv}} = 0.8$. The function $\text{sg}(\cdot)$ denotes the stop-gradient operation, and the weight β is set to 0.25.

In the first 400,000 steps of this stage, we use randomly cropped 256×256 images. Subsequently, we switch to 512×512 images for the next 400,000 steps of training. The enlarged training image size enables the model to be aware of cross-window interactions. In this stage, the λ values are set to $\{5.8, 8.5, 16.0, 28.0\}$ for different bitrates, and the model is trained with a learning rate of 2.0×10^{-5} and a batch size of 8.

B. Model Architectures

We illustrate the model architecture with the detailed hyperparameters. Fig. 1 shows the architecture of the semantic transform, the detail transform and the latent adaptor. Fig. 2 illustrates the downsample module before the scalar quantization and the upsample module after the quantization within the detail branch. Lastly, Fig. 3 depicts the design of the Interactive Transform (IT) module.

In DLF, we employ a non-overlapping window partitioning scheme to avoid additional bitrate overhead. In future work, it would be worthwhile to explore advanced overlapping window strategies (e.g., [20]) that improve compression efficiency without introducing redundancy.


W-MSA Block
dim=768, heads=12, mlp_dim=3072, window_size=16
SW-MSA Block
dim=768, heads=12, mlp_dim=3072, window_size=16
ConvNeXT Block
dim=768, kernel_size=5, mlp_dim=1536
ConvNeXT Block
dim=768, kernel_size=5, mlp_dim=1536
ViT Block
dim=1024, heads=16, mlp_dim=4096
ViT Block
dim=1024, heads=16, mlp_dim=4096
ViT Block
dim=1024, heads=16, mlp_dim=4096
ViT Block
dim=1024, heads=16, mlp_dim=4096
Concat
Attn Block
MLP
in_dim=1024, out_dim=256
MLP
in_dim=1792, out_dim=1024
W-MSA Block
dim=1024, heads=16, mlp_dim=4096, window_size=16
SW-MSA Block
dim=1024, heads=16, mlp_dim=4096, window_size=16
× 3

(a) Detail transform

(b) Semantic transform

(c) Latent adaptor

Figure 1. Hyper-parameter settings for the detail and semantic transform blocks, as well as the latent adaptor.

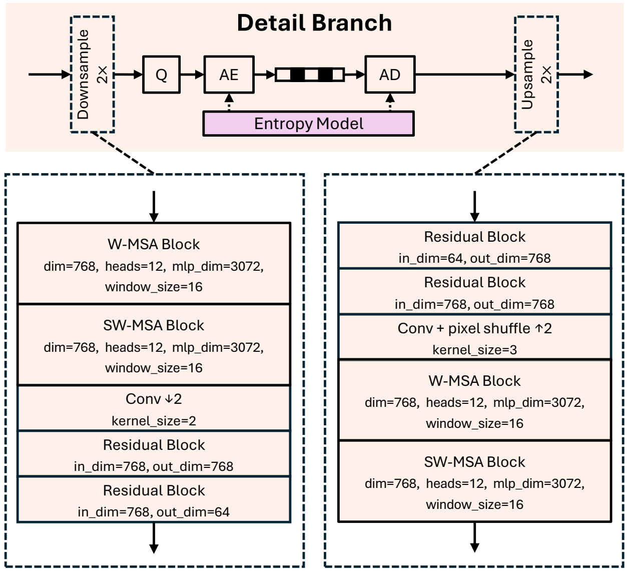


Figure 2. The structures and hyper-parameters of the downsample and upsample modules within the detail branch.

C. Experiments

C.1. Evaluation details

Evaluation of third-party models. We evaluate TCM [14], HiFiC [16], and MS-ILLM [17] by utilizing their official codes and fine-tuning their pretrained models (at the lowest bitrate) to achieve extremely low bitrate ranges. For DiffEIC [11], we employ their released models for evaluation. In the case of PerCo [3], we rely on a third-party implementation [9] for evaluation due to the absence of official code. Similarly, GLC [6] and RDEIC [12] also lack public code. For GLC, we obtain the evaluation results through the personal communication with the authors. For RDEIC and HybridFlow [15], we derive the number from their published papers, since there are no code or data available.

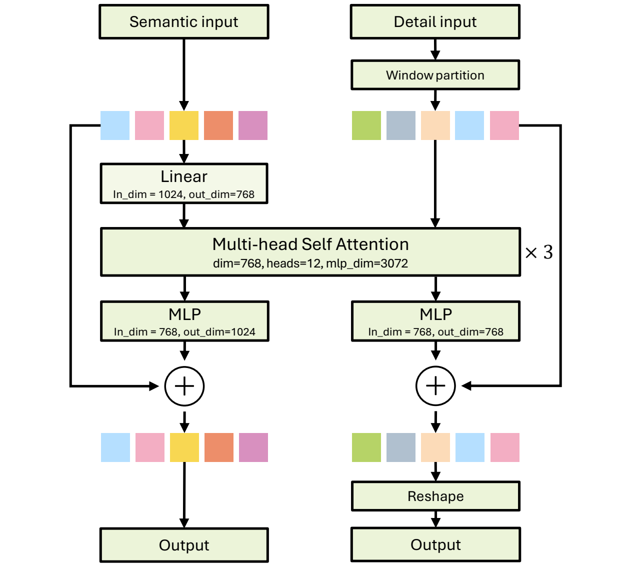

Semantic input
Detail input
Window partition
Linear
In_dim = 1024, out_dim=768
Multi-head Self Attention
dim=768, heads=12, mlp_dim=3072
× 3
MLP
In_dim = 768, out_dim=1024
MLP
In_dim = 768, out_dim=768
Reshape
Output
Output

Figure 3. Hyper-parameter settings for the Interactive Transform (IT) block.

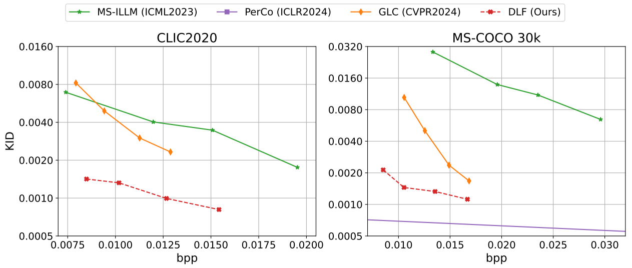


Figure 4. Rate-KID curves on the CLIC2020 and the MS-COCO 30K datasets.

Measurement of FID and KID. For the CLIC2020 test

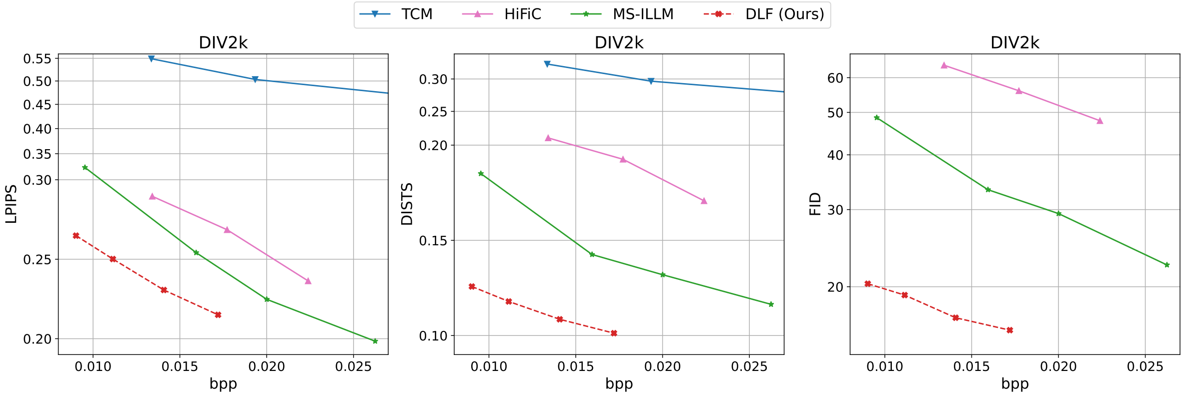


Figure 5. Comparison of methods on the DIV2K dataset [1].

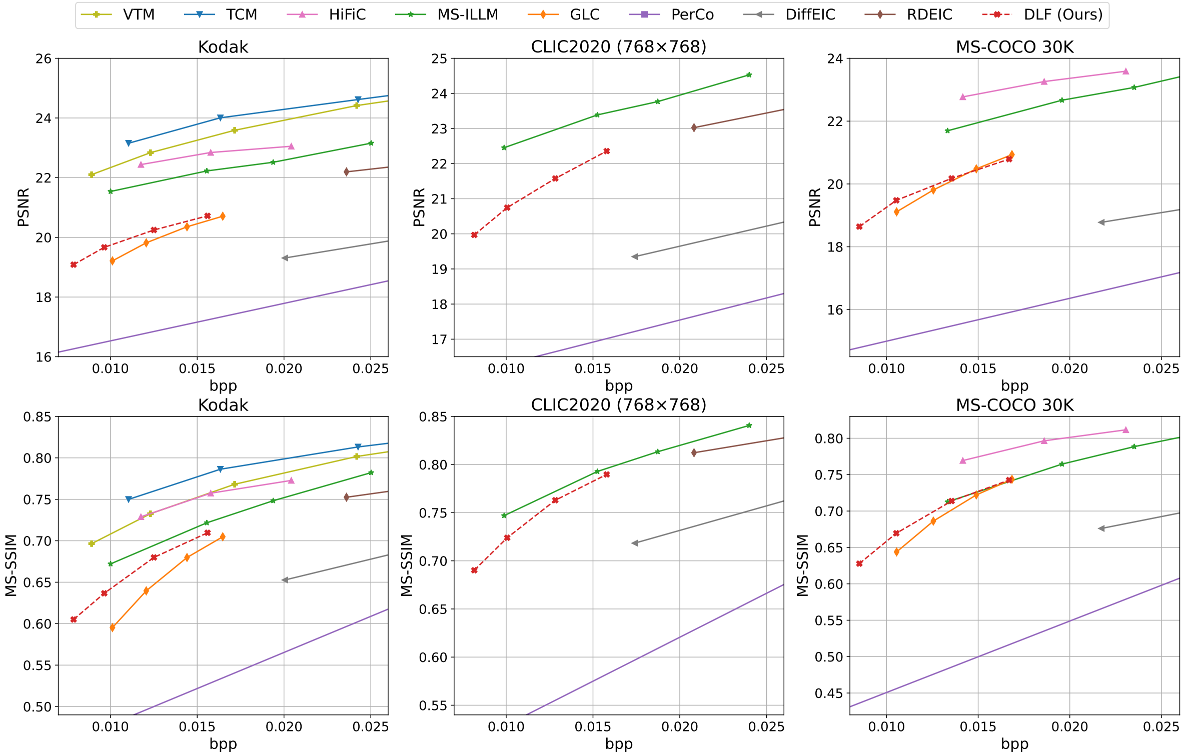


Figure 6. Comparison of methods measured by PSNR and MS-SSIM.

dataset [18] with full resolution, the FID [5] and KID [2] metric is evaluated by splitting the images into overlapped 256×256 patches, following the method in HiFiC [16]. This setting is also applied to the CLIC2020 test dataset with 768×768 resolution, in accordance with the condition in DiffEIC [11]. For the MS-COCO 30K dataset [13], we directly evaluate the FID and KID on 512×512 resolution.

C.2. Quantitative Results

We provide the additional KID [2] results on the CLIC2020 and the MS-COCO 30K datasets on the Fig. 4. We present the evaluation results on the DIV2K dataset [1] at full resolution in Fig. 5. Large version of the rate-distortion curves is shown in Fig. 12. Diffusion-based methods [3, 11] are excluded from evaluation on this dataset due to their large memory requirements, which exceed the capacity of our GPU (A100 with 40GB memory). From Fig. 5, we can see that our method outperforms MS-ILLM [17] across all reference and no-reference perceptual metrics, demonstrating its effectiveness. Additionally, we evaluate traditional pixel-level distortion metrics, PSNR and MS-SSIM [19], for a more comprehensive analysis, as shown in Fig. 6. It is

Table 1. Complexity analysis with model param count.

Model	Params	Enc. Time (s)	Dec. Time (s)	BD-Rate
MS-ILLM	181M	0.064 ± 0.010	0.070 ± 0.011	0.00%
PerCo	1.3B + 3.8 B*	0.461 ± 0.017	2.443 ± 0.011	-4.02%
DiffEIC	1.4B	0.152 ± 0.014	4.093 ± 0.042	14.67%
DLF	1.2B	0.178 ± 0.015	0.252 ± 0.014	-67.82%

* Open-sourced PerCo includes an additional 3.8B BLIP2 caption model.

worth noting that at extreme low bitrates, pixel-level distortion becomes too severe (e.g., the PSNR of VTM drops below 25 dB), making these metrics less meaningful for evaluating visual quality. Although our model does not exhibit the best pixel-level distortion metrics, it still provides the most visually appealing reconstructions with high fidelity, as demonstrated in previous sections.

C.3. Qualitative Results

In this section, we provide more visual examples across the CLIC2020 [18] (Full resolution: Fig. 7, 8, 9; 768×768: Fig. 10) and MS-COCO 30K [13] (Fig. 11) datasets. From these examples, we can find that DLF achieves the best quality with the lowest bitrate cost.

C.4. Complexity Analysis

Table 1 summarizes the coding time, model size, and BD-Rate comparison on the Kodak dataset. Compared to diffusion-based codecs, DLF delivers superior compression performance while being considerably more efficient in both model complexity and runtime. Notably, DLF eliminates the need for iterative denoising, resulting in substantially faster decoding. These results highlight the effectiveness of our design in improving parameter efficiency and reducing computational overhead.

References

[1] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2017. 3

[2] Mikołaj Bińkowski, Danica J Sutherland, Michael Arbel, and Arthur Gretton. Demystifying mmd gans. *arXiv preprint arXiv:1801.01401*, 2018. 3

[3] Marlene Careil, Matthew J Muckley, Jakob Verbeek, and Stéphane Lathuilière. Towards image compression with perfect realism at ultra-low bitrates. In *The Twelfth International Conference on Learning Representations*, 2023. 2, 3

[4] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12873–12883, 2021. 1

[5] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2017. 3

[6] Zhaoyang Jia, Jiahao Li, Bin Li, Houqiang Li, and Yan Lu. Generative latent coding for ultra-low bitrate image compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 26088–26098, 2024. 2

[7] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, San Diega, CA, USA, 2015. 1

[8] Ivan Krasin, Tom Duerig, Neil Alldrin, Vittorio Ferrari, Sami Abu-El-Haija, Alina Kuznetsova, Hassan Rom, Jasper Uijlings, Stefan Popov, Andreas Veit, Serge Belongie, Victor Gomes, Abhinav Gupta, Chen Sun, Gal Chechik, David Cai, Zheyun Feng, Dhyanesh Narayanan, and Kevin Murphy. Openimages: A public dataset for large-scale multi-label and multi-class image classification. *Dataset available from https://github.com/openimages*, 2017. 1

[9] Nikolai Körber, Eduard Kromer, Andreas Siebert, Sascha Hauke, Daniel Mueller-Gritschneder, and Björn Schuller. Perco (sd): Open perceptual compression, 2024. 2

[10] 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 (CVPR)*, pages 22616–22626, 2023. 1

[11] Zhiyuan Li, Yanhui Zhou, Hao Wei, Chenyang Ge, and Jingwen Jiang. Towards extreme image compression with latent feature guidance and diffusion prior, 2024. 2, 3

[12] Zhiyuan Li, Yanhui Zhou, Hao Wei, Chenyang Ge, and Ajmal Mian. Diffusion-based extreme image compression with compressed feature initialization, 2024. 2

[13] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014. 3, 4

[14] 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 (CVPR)*, pages 14388–14397, 2023. 2

[15] Lei Lu, Yanyue Xie, Wei Jiang, Wei Wang, Xue Lin, and Yanzhi Wang. Hybridflow: Infusing continuity into masked codebook for extreme low-bitrate image compression. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 3010–3018, 2024. 2

[16] Fabian Mentzer, George D Toderici, Michael Tschannen, and Eirikur Agustsson. High-fidelity generative image compression. In *Advances in Neural Information Processing Systems*, pages 11913–11924. Curran Associates, Inc., 2020. 2, 3

[17] Matthew J. Muckley, Alaaeldin El-Nouby, Karen Ullrich, Herve Jegou, and Jakob Verbeek. Improving statistical fidelity for neural image compression with implicit local likelihood models. In *Proceedings of the 40th International Conference on Machine Learning*, pages 25426–25443. PMLR, 2023. 2, 3

[18] George Toderici, Wenzhe Shi, Radu Timofte, Lucas Theis, Johannes Balle, Eirikur Agustsson, Nick Johnston, and

Fabian Mentzer. Workshop and challenge on learned image compression (clic2020), 2020. 3, 4

[19] Z. Wang, E.P. Simoncelli, and A.C. Bovik. Multiscale structural similarity for image quality assessment. In *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, pages 1398–1402 Vol.2, 2003. 3

[20] Qihang Yu, Xiaohui Shen, and Liang-Chieh Chen. Towards open-ended visual recognition with large language models. In *European Conference on Computer Vision*, pages 359–376. Springer, 2024. 1

[21] Qihang Yu, Mark Weber, Xueqing Deng, Xiaohui Shen, Daniel Cremers, and Liang-Chieh Chen. An image is worth 32 tokens for reconstruction and generation, 2024. 1

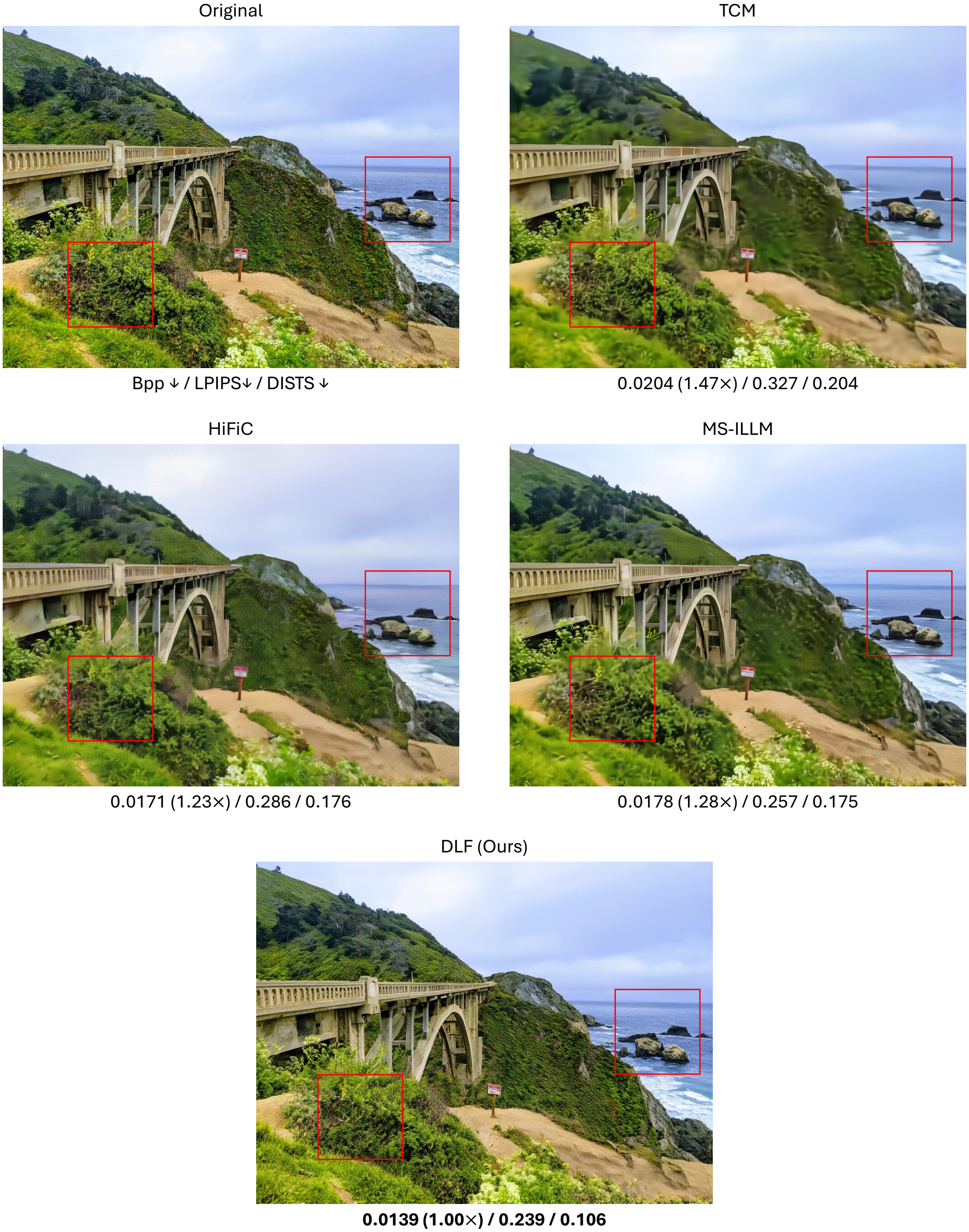


Figure 7. Qualitative examples on the CLIC2020 dataset (full resolution).

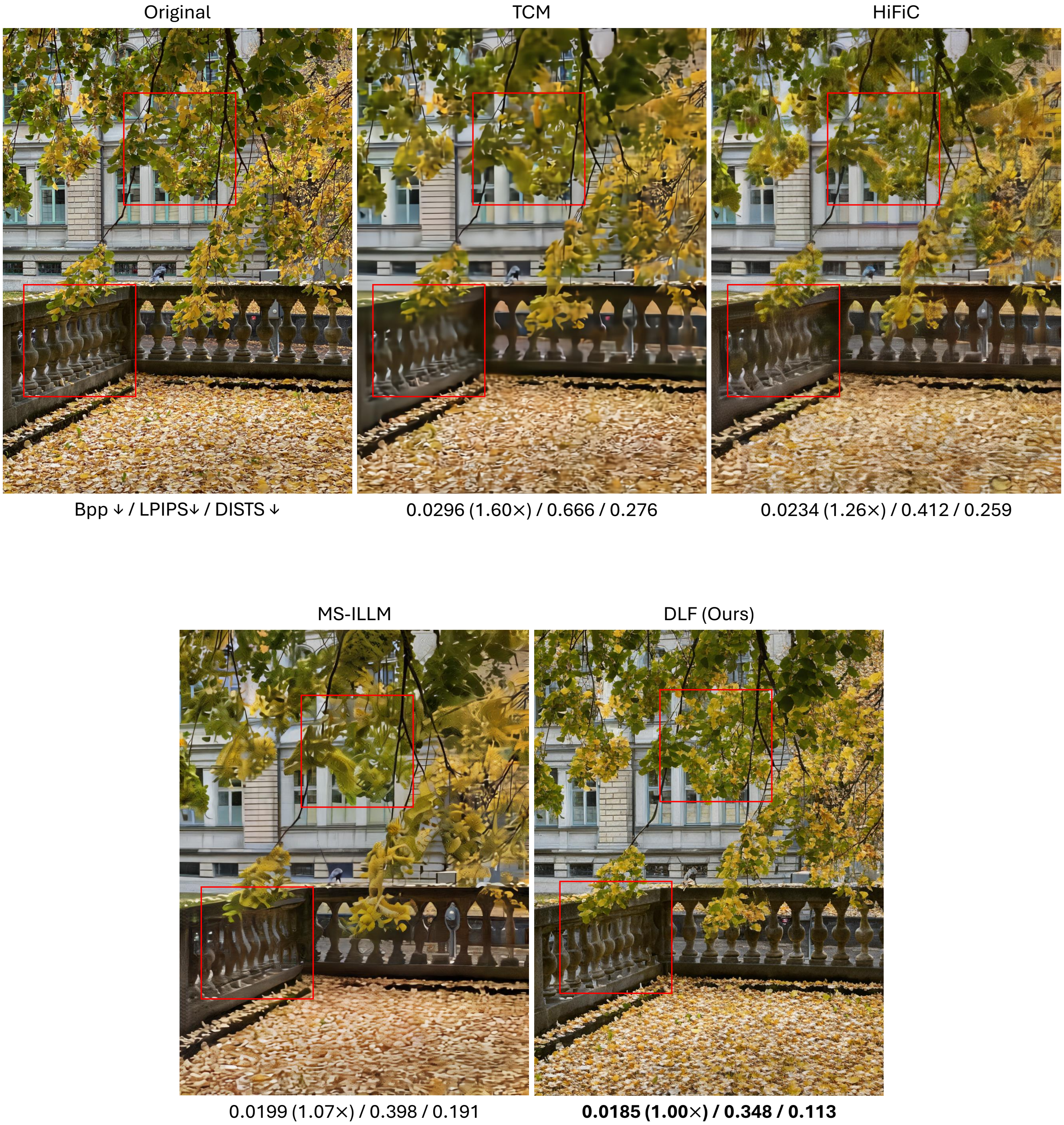


Figure 8. Qualitative examples on the CLIC2020 dataset (full resolution).

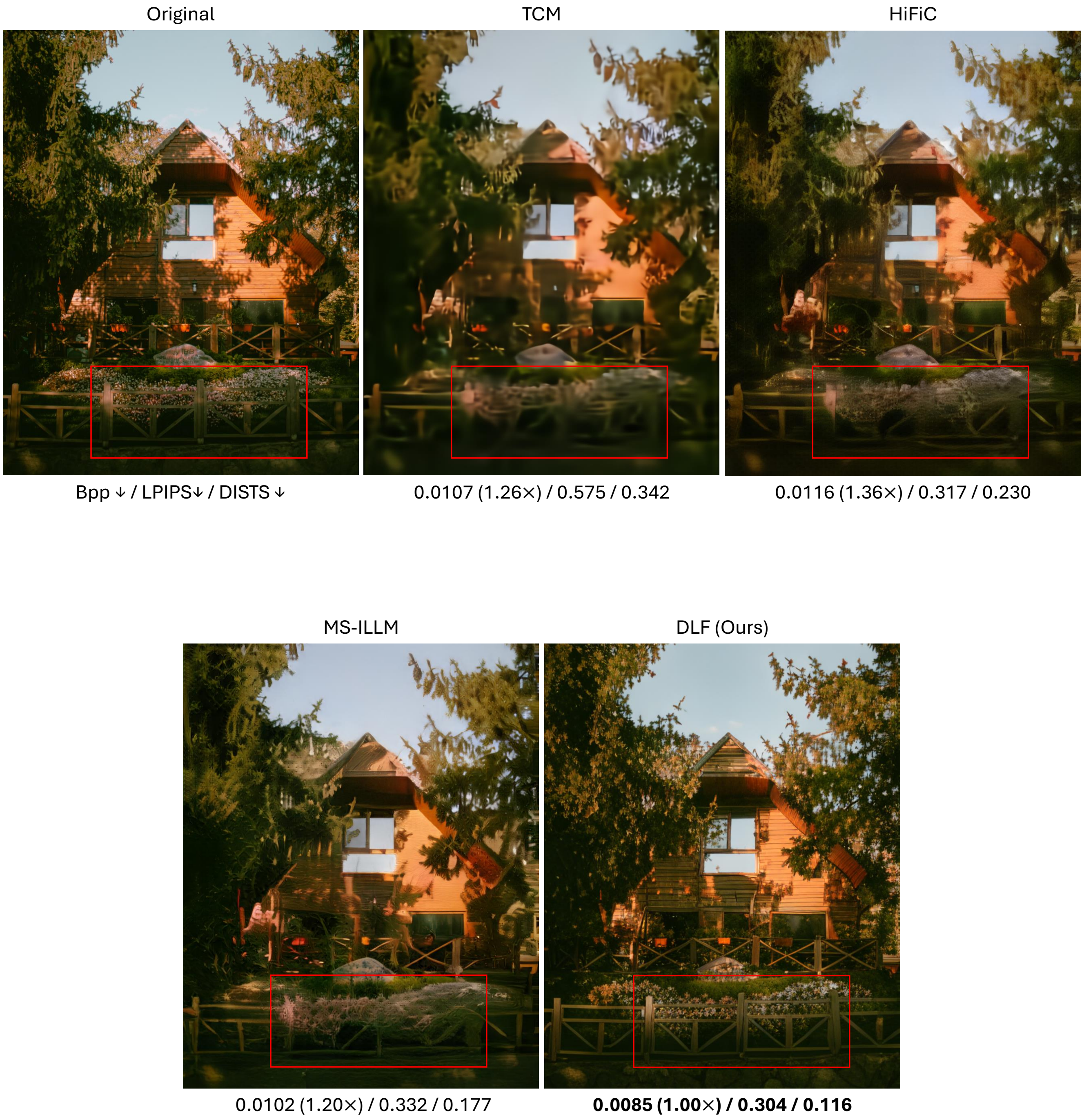


Figure 9. Qualitative examples on the CLIC2020 dataset (full resolution).

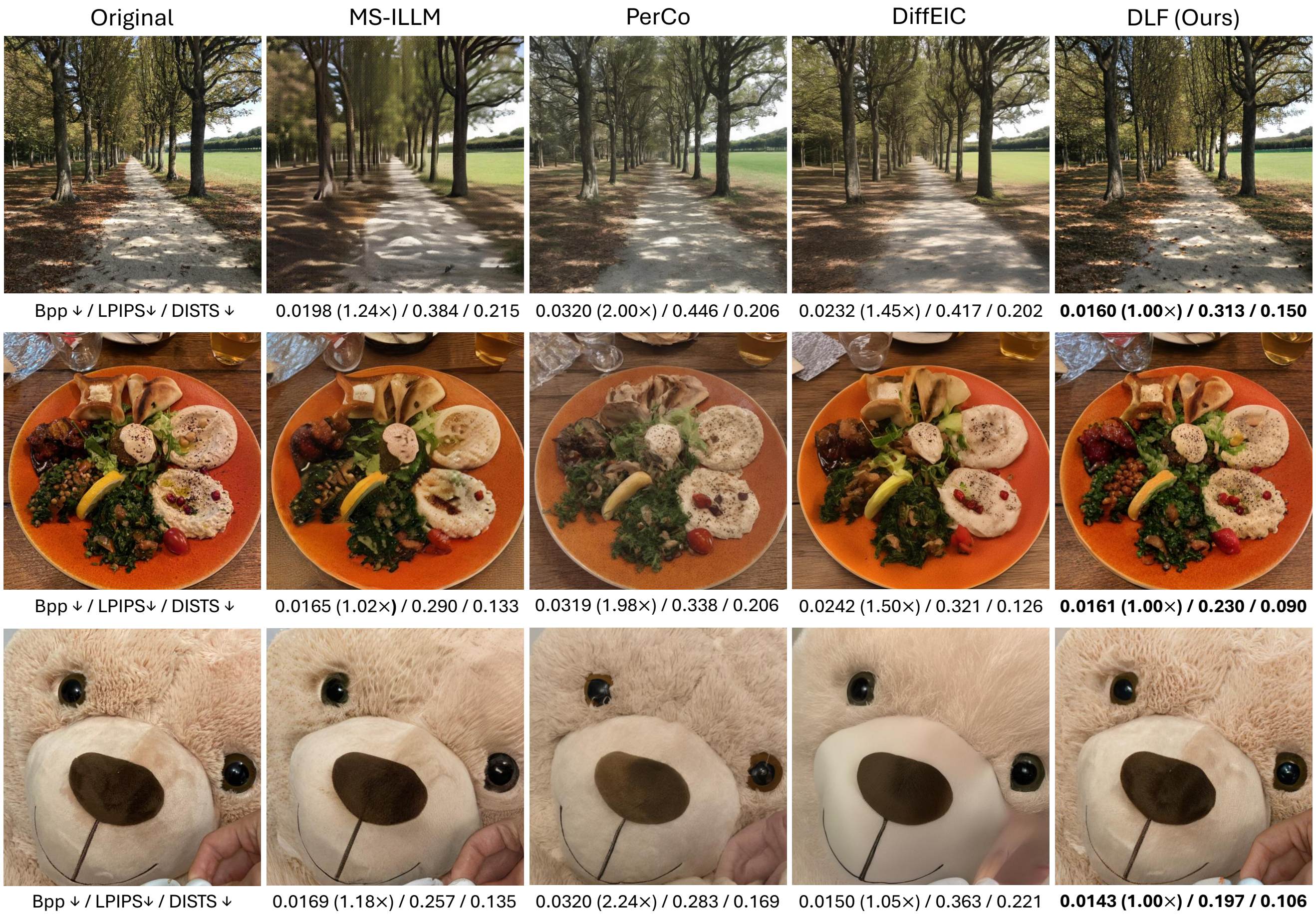


Figure 10. Qualitative examples on the CLIC2020 dataset (768×768).

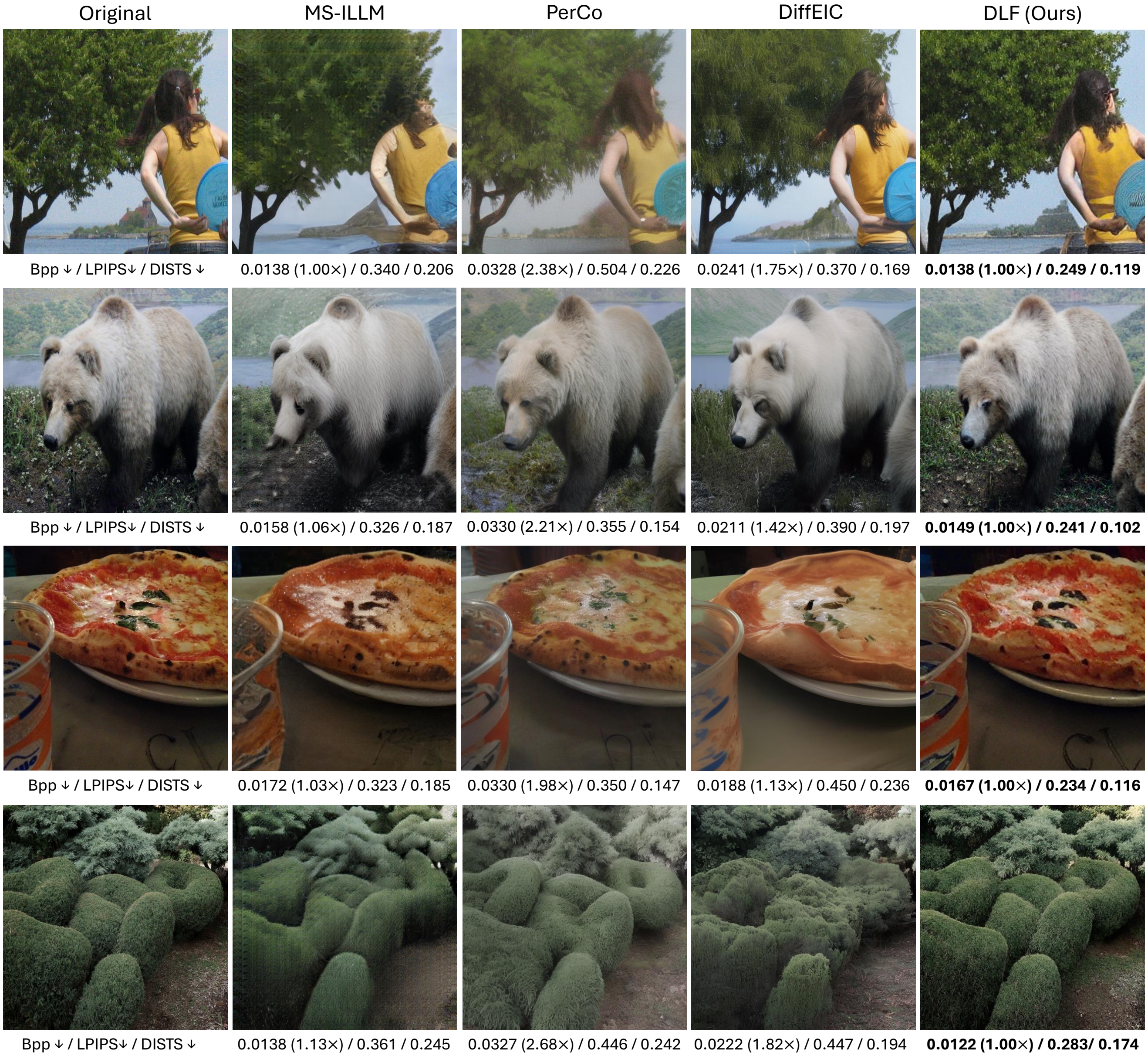


Figure 11. Qualitative examples on the MS-COCO 30K dataset.

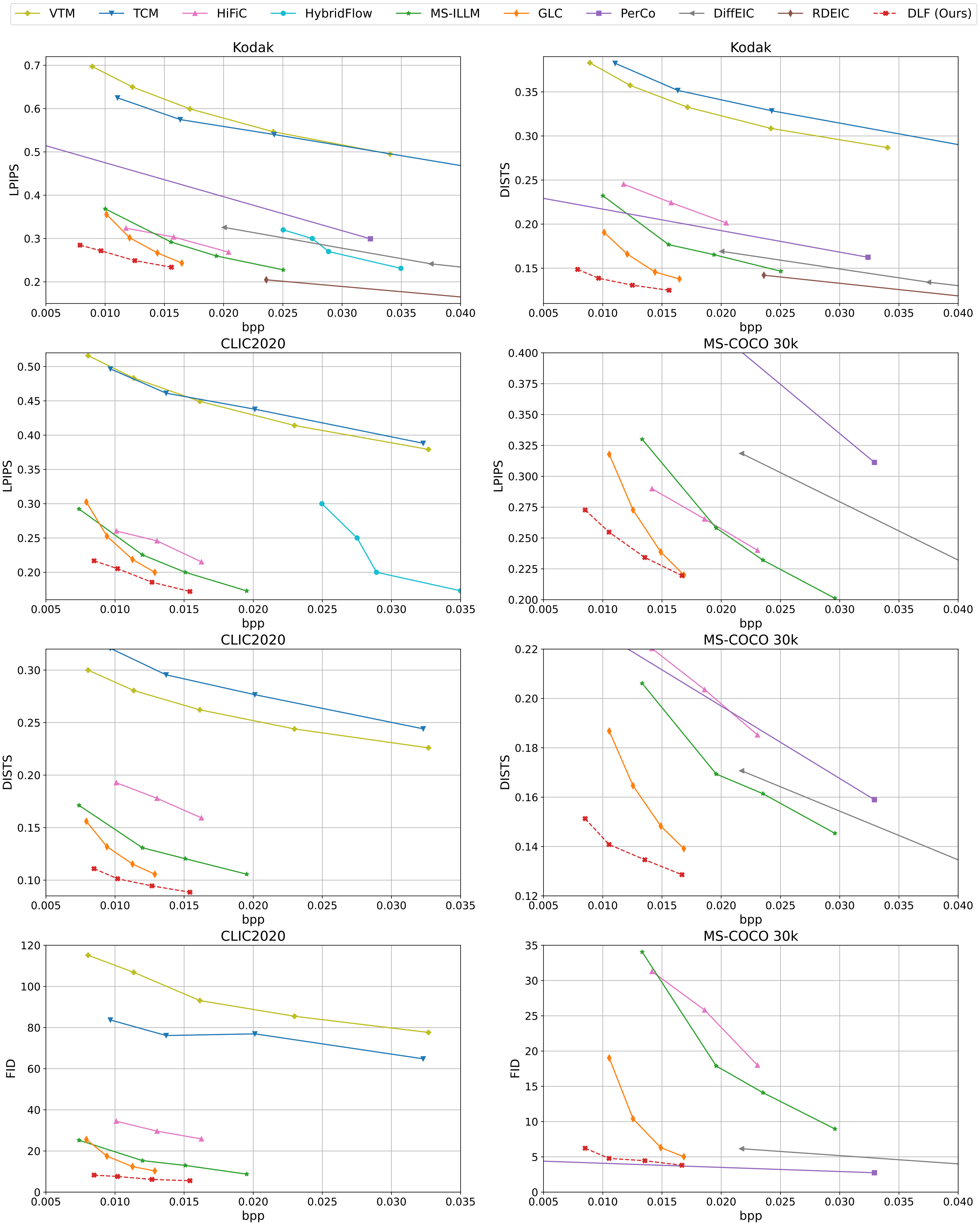


Figure 12. Rate-distortion curves on the Kodak, the CLIC2020 and the MS-COCO 30K datasets.