JDEC: JPEG Decoding via Enhanced Continuous Cosine Coefficients

Woo Kyoung Han[1,2] Sunghoon Im[2] Jaedeok Kim[3*] Kyong Hwan Jin[1*]
[1]Korea University [2]DGIST [3]NVIDIA
{wookyoung0727, kyong_jin}@korea.ac.kr, sunghoonim@dgist.ac.kr, jaedeokk@nvidia.com

Abstract

We propose a practical approach to JPEG image decoding, utilizing a local implicit neural representation with continuous cosine formulation. The JPEG algorithm significantly quantizes discrete cosine transform (DCT) spectra to achieve a high compression rate, inevitably resulting in quality degradation while encoding an image. We have designed a continuous cosine spectrum estimator to address the quality degradation issue that restores the distorted spectrum. By leveraging local DCT formulations, our network has the privilege to exploit dequantization and upsampling simultaneously. Our proposed model enables decoding compressed images directly across different quality factors using a single pre-trained model without relying on a conventional JPEG decoder. As a result, our proposed network achieves state-of-the-art performance in flexible color image JPEG artifact removal tasks. Our source code is available at https://github.com/WooKyoungHan/JDEC.

1. Introduction

Within the dynamic evolution of high-efficiency image compression, it is notable that JPEG [33] maintains a pivotal position. JPEG, renowned for its compatibility and standardization, is the most famous image coder-decoder (CODEC) among conventional lossy compression methods. Therefore, a high-quality JPEG decoder applies to all existing compressed JPEG files. JPEG reduces file size through downsampling color components and quantizing the discrete cosine transform (DCT) spectra, which leads to a complicated loss of image information and distortion. Consequently, the design of a high-quality JPEG decoder presents a dual challenge: 1) the restoration of complex losses from the JPEG encoder and 2) the modeling of a network that employs a spectrum as an input and its image as an output.

Many deep neural networks (DNNs) have been proposed as promising solutions for the JPEG artifact removal [11, 14, 18, 24, 35–37]. Most existing methods, such as [11, 24], are dedicated to specific quality factors, providing multiple models to cover JPEG compression. In recent studies [14, 18], the quality-dedicated problem has been addressed through the utilization of quantization maps [14] or the estimation of quality factors [18]. The existing artifact removal networks commonly take the decoded image as input, even though the encoded spectrum contains more information than the decoded image, according to the data processing inequality [10]. The property is explained in the supplement material.

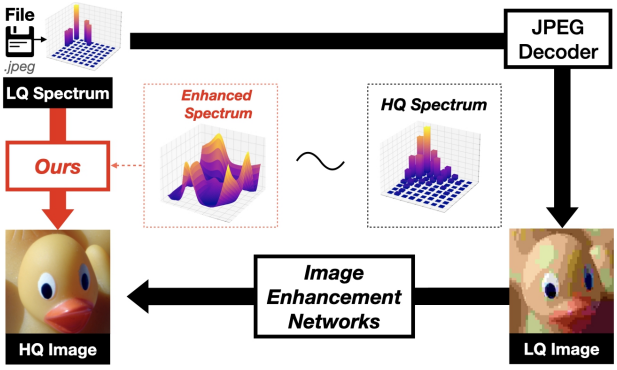


Figure 1. **Overall concept of proposed JPEG decoding** Instead of using a conventional JPEG decoder to refine the high-quality (HQ) image from the low-quality (LQ) image, our JDEC directly decodes the LQ spectrum by learning a continuous spectrum.

Due to the characteristics of the JPEG algorithm, it is non-trivial to design a neural network that takes spectra as inputs [15, 34]. Park *et al*. [28] proposed a method of processing spectra to transformers. In the context of spectral processing, our approach extends beyond proposed classification networks [15, 28, 34] by leveraging the capabilities of the embedding strategy, paving the way for more effective decoding. The spectrum conversion aligns with recent advancements in implicit neural representation (INR), where methods adopting sinusoidal functions [5, 16, 20, 21, 27, 32] have demonstrated significant advancement across various tasks.

In this paper, we propose an advanced model, the JPEG Decoder with Enhanced Continuous cosine coefficients (JDEC), for retrieving high-quality images from

*Corresponding author.

Figure 2. **Visual Demonstration at** $q = 100$ (PSNR (dB) ↑ / Bit-Error-Rate (BER) ↓) of decoding compressed image: JPEG (quality factor = 100), image enhancement approach [18] predicted from JPEG image ($q = 100$), and JDEC (*ours*) predicted directly from a JPEG bit-stream. We highlight the occurrence of bit errors overlaid with green dots.

compressed spectra. As an artifact removal network, our JDEC does not require a conventional JPEG decoder compared to existing methods shown in Fig. 1. JDEC captures the dominant frequency and its amplitude, thereby representing the high-quality spectrum through continuous cosine formulation (CCF). The CCF module estimates a continuous form of a given discrete cosine spectrum. The proposed model represents a considerable improvement in decoding JPEG bitstream. As shown in Fig. 2, our JDEC decodes high-quality images with fewer bit errors than the original JPEG decoder.

In summary, our main contributions are as follows:

- We propose a local implicit neural representation that decodes JPEG files across various quality factors (QF) with continuous cosine spectra.
- We show that the suggested continuous cosine formulation module lets the network predict spectra highly correlated with the ground truth's spectrum.
- We demonstrate that our proposed method operates as a practical decoder, delivering superior image quality, including the generally used quality factor.

2. Related Work

JPEG Background According to Shannon's source coding theorem [30], a loss of image information is unavoidable to achieve high-efficiency compression. The JPEG initiates the encoding process by decomposing an input RGB image to luminance and chroma components [33]. The chroma components are downsampled using the nearest neighbor method by a factor of $\times 2$. The JPEG subtracts the midpoint of the pixel value (=128) to images and divides it into 8×8 crops. Then, each crop is transformed into 2D-DCT [2] spectra. Following this, the encoder quantizes the spectrum of each block using a predefined quantization matrix depending on a quality factor q, and then the quantized spectrum is coded using Huffman coding. We illustrate the process of the JPEG encoder in Fig. 3.

Due to the nature of DCT, the energy of spectra is concentrated in low-frequency components. Since the quantization matrix treats high-frequency components more severely than low-frequency components, most distortions

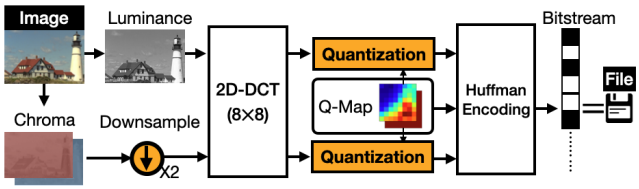


Figure 3. **Overall process of the JPEG encoder.** Luminance and chroma components are separated from an RGB image. Both components are converted to DCT spectra and quantized with a pre-defined quantization matrix (Q-map). All losses occur in the orange area.

occur in the high-frequency components. In the JPEG decoder, the quantization matrix is directly applied to the quantized spectra, transforming them into images. Consequently, all incurred losses, especially those in high-frequency components, are directly conveyed to the resulting image.

JPEG Artifact Removal To address the aforementioned problem, learning-based methods have enhanced the quality of a decoded image. Dong *et al*. [11] introduced a neural network that utilizes a super-resolution network [12] for JPEG artifact removal. Most of the proposed neural networks are dedicated to a specific quality factor [7, 8, 11, 23]. To tackle the quality-dedicated issue, Jiang *et al*. [18] proposed a method to estimate a quality factor, solving flexible JPEG artifact removal and handling a double JPEG artifact. However, the existing artifact removal methods take images as input, incorporating the conventional JPEG decoder before using their network. Recently, Bahat *et al*. [4] proposed a novel method for JPEG decoding, which takes spectra as input. However, the proposed method does not consider color components with trainable decoding and does not recover high-quality factors.

Learning in the Frequency domain In an image classification task, skipping a conventional JPEG decoding [15, 34] has been proposed, especially optimizing CNNs. Embedding techniques [15] tackle the size mismatch issue between luma and chroma components, such as upsampling chroma components before forwarding to a network and upsampling chroma features after forwarding to a shallow network. The proposed methods boost computation time without dropping the original performance. Recently, the approach adopting vision transformers [13] instead of CNNs has promising performance [28]. We adopt the proposed embedding method from [28] and modified [25] SwinV2 transformer suitable for image decoding.

3. Method

Problem Formulation Let $\mathbf{I_{GT}} \in \mathbb{R}^{H \times W \times 3}$ be a ground-truth RGB image. The JPEG encoder separates $\mathbf{I_{GT}}$ to luminance component ($\mathbf{I_Y} \in \mathbb{R}^{H \times W \times 1}$) and chroma components ($\mathbf{I_C} \in \mathbb{R}^{H \times W \times 2}$) and downsamples chroma components by a factor of 2, i.e. ($\mathbf{I_C^{\downarrow}} \in \mathbb{R}^{\frac{H}{2} \times \frac{W}{2} \times 2}$). The super-

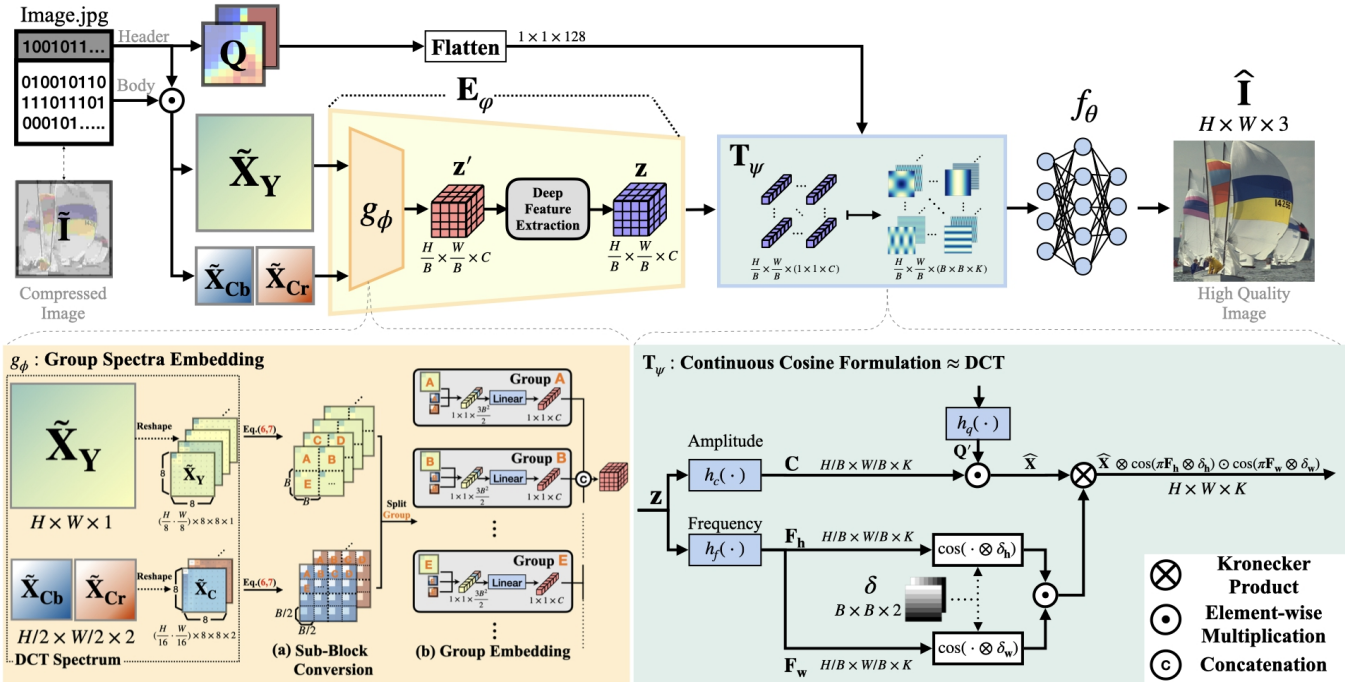


Figure 4. **Decoding a JPEG bitstream with the proposed JDEC.** JDEC consists of an encoder (E_φ) with group spectra embedding (g_ϕ), a decoder (f_θ), and continuous cosine formulation (T_ψ). Inputs of JDEC are as follows: compressed spectra ($\tilde{\mathbf{X}}_\mathbf{Y}, \tilde{\mathbf{X}}_\mathbf{C}$), quantization map $\mathbf{Q}$. Note that our JDEC does not take $\tilde{\mathbf{I}}$ as an input. JDEC formulates latent features into a trainable continuous cosine coefficient as a function of block grid δ and forward to INR (f_θ). Therefore, each $B \times B$ block shares the estimated continuous cosine spectrum.

script $\downarrow$ indicates a $\times 2$ downsampling. Then, each component is divided into 8×8 blocks ($\mathbf{I} \in \mathbb{R}^{8\times 8} \subset \mathbf{I}_\mathbf{Y}, \mathbf{I}_\mathbf{C}^\downarrow$). 2D-DCT [2] into spectra $\mathbf{X} \in \mathbb{R}^{8\times 8}$ is defined as below:

$$DCT(\mathbf{I}) := \mathbf{X} = \mathbf{D}\mathbf{I}\mathbf{D}^\top. \tag{1}$$

The orthonormal basis matrix $\mathbf{D}(=\mathbf{D}_8)$ is defined as:

$$\mathbf{D}_N := [\alpha] \odot \cos([\pi[F_{k|N}]_{k=0}^{N-1} \otimes [k]_{k=0}^{N-1^\top}]) \tag{2}$$
$$= \sqrt{\frac{2}{N}} \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \cdots & \frac{1}{\sqrt{2}} \\ \cos\left(\frac{1\pi}{2N}\right) & \cos\left(\frac{3\pi}{2N}\right) & \cdots & \cos\left(\frac{(2N-1)\pi}{2N}\right) \\ \vdots & \vdots & \ddots & \vdots \\ \cos\left(\frac{(N-1)\pi}{2N}\right) & \cos\left(\frac{3(N-1)\pi}{2N}\right) & \cdots & \cos\left(\frac{(2N-1)(N-1)\pi}{2N}\right) \end{bmatrix}.$$

where $F_{k|N} := (2k+1)/2N$ is a fixed frequency of a coordinate k with a given size N and $[\alpha]$ is the scaling matrix for orthonormality. The operations $\otimes$ and $\odot$ are a Kronecker product and element-wise multiplication, respectively.

Quantization is conducted with a predefined quantization matrix $\mathbf{Q} = [\mathbf{Q}_\mathbf{Y}; \mathbf{Q}_\mathbf{C}] \in \mathbb{N}^{8\times 8\times 2}$ s.t.

$$\tilde{\mathbf{C}} := \left\lfloor \mathbf{X} \odot \frac{1}{\mathbf{Q}} \right\rceil, \ \tilde{\mathbf{X}} = \tilde{\mathbf{C}} \odot \mathbf{Q}, \tag{3}$$

where $\lfloor \cdot \rceil$ is a round operation that maps to the nearest integer. $1/\mathbf{Q}$ denotes an element-wise division. The JPEG encoder compresses the header $\mathbf{Q}$ and the body of a code $\tilde{\mathbf{C}}$ separately. In the decoder part, the JPEG restores the image from the compressed header and body by $\tilde{\mathbf{I}} = DCT^{-1}(\tilde{\mathbf{C}} \odot \mathbf{Q})$.

To summarize this, the corrupted JPEG image $\tilde{\mathbf{I}}$ is obtained by

$$\begin{bmatrix} \tilde{\mathbf{I}}_\mathbf{Y} \\ \tilde{\mathbf{I}}_\mathbf{C} \end{bmatrix} = \begin{bmatrix} DCT^{-1}(\lfloor DCT(\mathbf{I}_\mathbf{Y}) \odot \frac{1}{\mathbf{Q}_Y} \rceil \odot \mathbf{Q}_\mathbf{Y}) \\ DCT^{-1}(\lfloor DCT(\mathbf{I}_\mathbf{C}^\downarrow) \odot \frac{1}{\mathbf{Q}_C} \rceil \odot \mathbf{Q}_\mathbf{C})^\uparrow \end{bmatrix}, \tag{4}$$

where the superscript $\uparrow$ indicates $\times 2$ upsampling in the spatial domain. We here observe from Eq. (4) that most of the loss of image information is induced by the quantization step. The shape of the chroma component $\tilde{\mathbf{X}}_\mathbf{C} \in \mathbb{R}^{\frac{H}{2}\times\frac{W}{2}\times 2}$ is different from the luminance component $\tilde{\mathbf{X}}_\mathbf{Y} \in \mathbb{R}^{H\times W\times 1}$. We will consider observations in the design of our proposed network.

We propose a JPEG decoder network, JDEC $\mathbf{J}_\Theta$, defined by

$$\mathbf{J}_\Theta : (\tilde{\mathbf{X}}_\mathbf{Y}, \tilde{\mathbf{X}}_\mathbf{C}; \mathbf{Q}) \mapsto \hat{\mathbf{I}}. \tag{5}$$

The network directly accepts quantized spectrum $\tilde{\mathbf{X}}_\mathbf{Y}$ and $\tilde{\mathbf{X}}_\mathbf{C}$ with a quantization matrix as the network inputs, enabling to decode JPEG directly from encoded JPEG data. Our proposed JDEC comprises mainly three parts: encoder $\mathbf{E}_\varphi$ with group embedding, continuous cosine formulation $\mathbf{T}_\psi$, and decoder f_θ with an implicit neural representation.

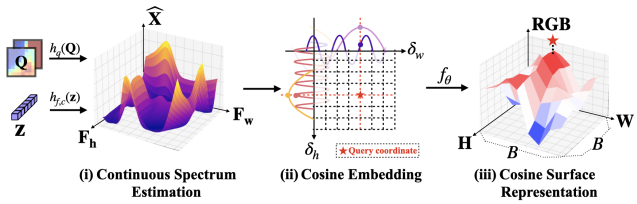


Figure 5. **Graphical summary of $f_\theta(T_\psi(\delta, \mathbf{z}; \mathbf{Q}))$.** Each 1×1-sized feature $\mathbf{z}$ maps into a $B \times B$ pixel area. T_ψ embeds the local coordinates of $B \times B$ area and forwards to f_θ.

Encoder ($\mathbf{E}_\varphi$) The encoder is a function $\mathbf{E}_\varphi \colon (\tilde{\mathbf{X}}_\mathbf{Y}, \tilde{\mathbf{X}}_\mathbf{C}) \mapsto \mathbf{z} \in \mathbb{R}^{\frac{H}{B} \times \frac{W}{B} \times C}$. We model the encoder $\mathbf{E}_\varphi$ by using SwinV2 [25]. To allow the different shape of spectrum $\tilde{\mathbf{X}}_\mathbf{Y}$ and $\tilde{\mathbf{X}}_\mathbf{C}$, we apply the group spectra embedding layer g_ϕ proposed by [28]. The embedding layer (g_ϕ) transforms luminance and chroma spectra through two steps. We convert 8×8 spectra into $B \times B$ for luma and $B/2 \times B/2$ for chroma via sub-block conversion [17] in the (a) part of g_ϕ in Fig. 4. We implement the block size $B = 4$[1].

$$\mathbf{X}'_\mathbf{Y} = \mathbf{D}^*_B(\mathbf{D}^\top \mathbf{X}_\mathbf{Y} \mathbf{D})\mathbf{D}^{*\top}_B, \tag{6}$$

$$\mathbf{X}'_\mathbf{C} = \mathbf{D}^*_{B/2}(\mathbf{D}^\top \mathbf{X}_\mathbf{C} \mathbf{D})\mathbf{D}^{*\top}_{B/2}, \tag{7}$$

where D^*_N indicates a block diagonal matrix with size 8×8. In part (b) of Fig. 4, spectra are reshaped and concatenated to $\mathbb{R}^{\frac{H}{B} \times \frac{W}{B} \times \frac{3B^2}{2}}$ which is the sum of converted size. Then initialized latent vector $\mathbf{z}' \in \mathbb{R}^{\frac{H}{B} \times \frac{W}{B} \times C}$ are conducted in (b) part of g_ϕ. Following the prior work, [22], we adopt the deep feature extractor, replacing the Swin attention module with the SwinV2 attention module.

Continuous Cosine Formulation ($\mathbf{T}_\psi$) Each JPEG block shares a distorted DCT spectrum. Modifying the entire spectrum is required to restore the distortion of a block. To address the spectrum distortion issue derived from the JPEG encoder, we introduce *Continuous Cosine Formulation* (CCF) module, which enhances the cosine spectrum. The CCF constructs a continuous spectrum corresponding to $B \times B$ embedded block by estimating dominant frequencies and amplitudes of a cosine transform. Illustrated in Figure 5, each block has identical amplitudes and frequencies within the embedded block coordinate $\delta := [(i, j)]^{B-1}_{i,j=0}$.

Our CCF takes a latent vector $\mathbf{z}$ from encoder $\mathbf{E}_\varphi$ and a quantization matrix $\mathbf{Q}$. The CCF $\mathbf{T}_\psi$ consists of three elements: frequency estimator $h_f \colon \mathbb{R}^C \mapsto \mathbb{R}^{2K}$, coefficient estimator $h_c \colon \mathbb{R}^C \mapsto \mathbb{R}^K$, and quantization matrix encoder $h_q \colon \mathbb{R}^{128} \mapsto \mathbb{R}^K$. Each frequency and coefficient estimator comprises sequential convolution and non-linear activation layers. As a method for quantization recovery, we implement an amplitude recovery method as described below, drawing inspiration from the existing dequantization network [16],

$$\widehat{\mathbf{X}} = \mathbf{C} \odot \mathbf{Q}' \sim Eq.\ (3), \tag{8}$$

where $\mathbf{Q}' = h_q(\mathbf{Q})$ and $\mathbf{C} = h_c(\mathbf{z})$.

We hypothesize that estimating the frequency components effectively mitigates aliasing (i.e., quantization and downsampling) derived from JPEG. It has been demonstrated that trainable frequencies and phasors effectively mitigate upsampling and dequantization [16, 21].

We thus formulate the CCF module approximates $B \times B$ spectral features from the fiber of $\mathbf{z}$:

$$\mathbf{T}_\psi(\mathbf{z}, \delta_{\mathbf{h},\mathbf{w}}; \mathbf{Q}) = \widehat{\mathbf{X}} \otimes (\cos(\pi \mathbf{F_h} \otimes \delta_\mathbf{h}) \odot \cos(\pi \mathbf{F_w} \otimes \delta_\mathbf{w})), \tag{9}$$

where $[\mathbf{F_h}; \mathbf{F_w}] = h_f(\mathbf{z})$. $\delta_{\mathbf{h},\mathbf{w}}$ denotes vertical and horizontal coordinates of δ. Note that $\widehat{\mathbf{X}}, \mathbf{F_h}, \mathbf{F_w} \in \mathbb{R}^K$ are amplitude and frequencies for the spatial coordinate δ, respectively. i.e. the CCF maps embedded features and block coordinates by $\mathbf{T}_\psi : (\mathbb{R}^{1\times1\times C}, \mathbb{R}^{B\times B\times 2}) \mapsto \mathbb{R}^{B\times B\times K}$.

Decoder (f_θ) Our decoder $f_\theta \colon \mathbb{R}^K \mapsto \mathbb{R}^3$ is a local implicit neural representation function of $\{\mathbf{z}, \mathbf{Q}\}, \delta$. i.e. :

$$\widehat{\mathbf{I}} = f_\theta(\widehat{\mathbf{X}} \otimes (\cos(\pi \mathbf{F_h} \otimes \delta_\mathbf{h}) \odot \cos(\pi \mathbf{F_w} \otimes \delta_\mathbf{w}))). \tag{10}$$

Therefore, in the $B \times B$ block of Eq. (10), the estimated basis of $\widehat{\mathbf{X}}$ and its reconstruction follows:

$$\mathbf{I} = \mathbf{D}^\top \mathbf{X} \mathbf{D} \simeq f'_\theta(\mathbf{\Lambda_h} \widehat{\mathbf{X}}' \mathbf{\Lambda_w}), \tag{11}$$

where $\mathbf{\Lambda}_{\mathbf{h},\mathbf{w}} = \cos([\pi \mathbf{F}_{\mathbf{h},\mathbf{w}} \otimes \delta_{\mathbf{h},\mathbf{w}}])$ and f'_θ satisfy $f_\theta = f'_\theta \circ W$ for a trainable fully-connected layer W. With a linear layer W, the Eq. (10) complete the quadratic form $\mathbf{\Lambda}_1 \widehat{\mathbf{X}}' \mathbf{\Lambda}_2 = W(\mathbf{T}_\psi(\mathbf{z}, \mathbf{Q}; \delta))$ by including summation of features. We optimize a set of trainable parameters $\Theta := \{\varphi; \psi; \theta\}$ with the equation below:

$$\widehat{\Theta} = \arg\min_\Theta ||\mathbf{I}_{GT} - \widehat{\mathbf{I}}(\tilde{\mathbf{X}}_\mathbf{Y}, \tilde{\mathbf{X}}_\mathbf{C}, \mathbf{Q}; \Theta)||_1. \tag{12}$$

We will demonstrate the estimated frequencies $(\mathbf{F_h}, \mathbf{F_w})$ and amplitudes $\widehat{\mathbf{X}}$ of networks follows $\mathbf{X}$ in the following section.

4. Experiments

4.1. Network Details

Encoder ($\mathbf{E}_\varphi$) and Decoder (f_θ) The linear layer in group spectra embedding module g_ϕ has an embedding size C of 256. We modified the deep feature extract part of SwinIR [22]. The window attention module is replaced with SwinV2 [25], with a window size of 7. [22] and [28] reported that a window size of 8 significantly drops the performance of the network. Each residual Swin transformer block includes 6 Swin transformer layers. The decoder f_θ is

[1] B should be divisor or multiple of 16

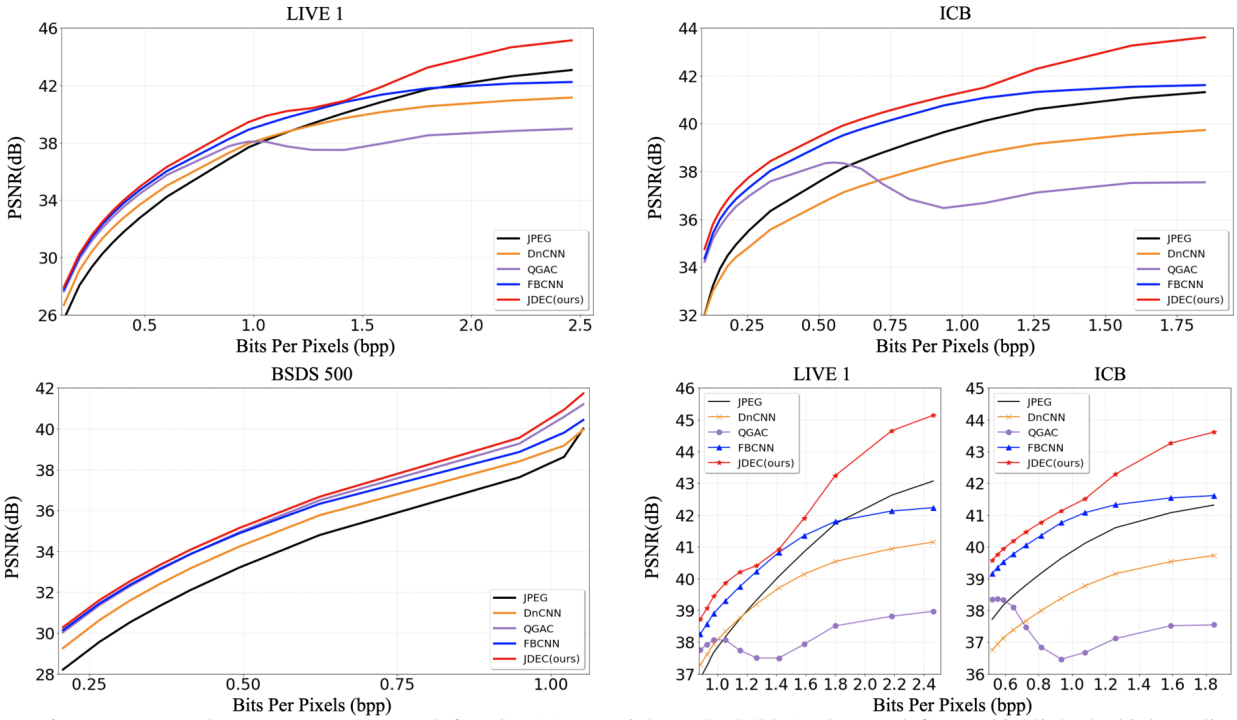


Figure 6. RD curve results on LIVE-1 [31] (top left), ICB [29] (top right), BSDS500 [3] (bottom left). We highlight the high-quality factor parts $q \in [90, 100]$ in the bottom right part. We show PSNR as a measure of *distortion* (higher is better). We observe that our JDEC decodes high-quality images better than other methods.

Test	LIVE-1 [31]				BSDS500 [3]				ICB [29]			
Method	$q=10$	$q=20$	$q=30$	$q=40$	$q=10$	$q=20$	$q=30$	$q=40$	$q=10$	$q=20$	$q=30$	$q=40$
JPEG	25.69|24.20 0.759	28.06|26.49 0.841	29.37|27.84 0.875	30.28|28.84 0.894	25.84|24.13 0.759	28.21|26.37 0.844	29.57|27.72 0.880	30.52|28.69 0.900	29.44|28.53 0.753	32.01|31.11 0.807	33.20|32.35 0.833	33.95|33.14 0.844
DMCNN [36]	27.18|27.03 0.810	29.45|29.08 0.874	-	-	27.16|26.95 0.799	29.35|28.84 0.866	-	-	30.85| - 0.796	32.77| - 0.830	-	-
IDCN [37]	27.62|27.32 0.816	30.01|29.49 0.881	-	-	27.61|27.22 0.805	28.01|25.57 0.873	-	-	31.71| - 0.809	33.99| - 0.838	-	-
Swin2SR* [9]	27.98| - -	-	-	32.53| - -	-	-	-	-	32.46| - -	-	-	36.25| - -
DnCNN [35]	26.68|26.47 0.794	29.12|28.77 0.866	30.43|30.04 0.895	31.34|30.94 0.911	26.82|26.53 0.793	29.26|28.74 0.867	30.63|30.02 0.898	31.59|30.92 0.915	29.78|29.71 0.726	31.99|31.90 0.765	32.98|32.89 0.786	33.52|33.42 0.978
QGAC [14]	27.65|27.43 0.819	29.88|29.56 0.882	31.17|30.77 0.908	32.08|31.64 0.922	27.75|27.48 0.819	30.04|29.55 0.884	31.36|30.73 0.911	32.29|31.53 0.926	32.12|32.09 0.814	34.22|34.18 0.844	35.18|35.13 0.859	35.71|35.65 0.865
FBCNN [18]	27.77|27.51 0.816	30.11|29.70 0.881	31.43|30.92 0.908	32.34|31.80 0.923	27.85|27.53 0.814	30.14|29.58 0.881	31.45|30.74 0.909	32.36|31.54 0.924	32.18|32.15 0.813	34.38|34.34 0.844	35.41|35.35 0.859	36.02|35.95 0.869
JDEC ***(ours)***	27.95|27.71 0.821	30.26|29.87 0.885	31.59|31.12 0.911	32.50|31.98 0.925	28.00|27.67 0.819	30.31|29.71 0.885	31.65|30.88 0.912	32.53|31.68 0.927	32.55|32.51 0.818	34.73|34.68 0.847	35.75|35.68 0.862	36.37|36.28 0.871

Table 1. Quantitative comparisons (PSNR (dB) | PSNR-B (dB) (top), SSIM (bottom)) with *the color JPEG artifact removal* networks. Red and blue colors indicate the best and the second-best performance, respectively. (-) indicates not reported. (*) indicates using additional datasets. Note that only JPEG [33] and our JDEC get spectra as input.

an MLP composed of 5 linear layers with 512 hidden channels K and ReLU activations.

CCF The CCF includes a frequency estimator h_f, an amplitude estimator h_c, and a quantization matrix encoder h_q. [16, 20, 21] show that learning frequency, phase, and amplitude components enhance the performance of the INR. The quantization matrix encoder h_q is a single fully connected layer, having $512(=K)$ channels. The amplitude and frequency estimator (h_c, h_f) is designed with two 3×3 convolutional layers with a ReLU activation. The frequency estimator has $2K(=1024)$ output channels for h and w axis, while the amplitude estimator has $K(=512)$ channels.

4.2. Training

Dataset Following the previous work [14, 18], we use DIV2K and Flickr2K [1]. Each dataset contains 800 and 2650 images, respectively. For generating synthetic JPEG compression, we use the OpenCV standard [6]. We compress images using randomly sampled quality factors with steps of 10 in the range [10,100]. We directly extract quantization maps $\mathbf{Q}$ and coefficients of spectra $\tilde{\mathbf{C}}$ from JPEG files and construct spectra $\tilde{\mathbf{X}}$, following the Eq. (3). Since

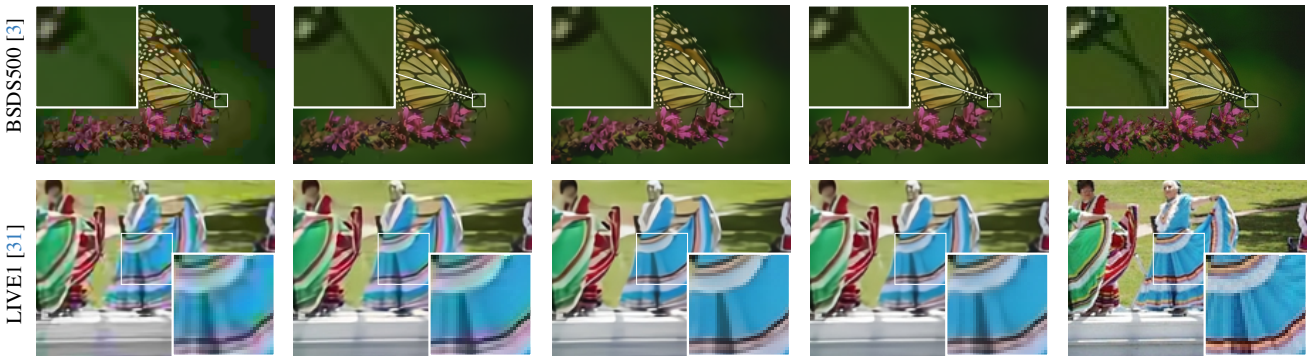


Figure 7. Qualitative comparison in color JPEG artifact removal ($q = 10$).

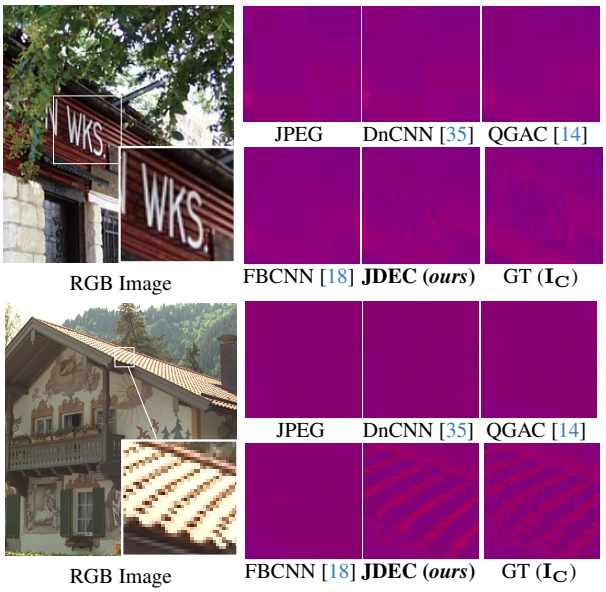


Figure 8. Qualitative comparison in chroma components $\mathbf{I_c}$ of images ($q = 10$).

the dynamic range of spectra depends on frequency, we should normalize spectra in a range of $[-1, 1]$. The quantization maps are normalized with the same normalization function. The ground truth (GT) images are prepared with a range of $[-0.5, 0.5]$ because the JPEG encoder subtracts the midpoint of the image range (=128).

Implementation Detail We use 112×112 patches as inputs to our network. This size is chosen because it is the least common multiple of the minimum unit size of color JPEG (16×16) and the window size of our Swin architecture [25] (7×7). The network is trained for 1000 epochs with batch size 16. We optimize our network by Adam [19]. The learning rate is initialized as 1e-4 and decayed by factor 0.5 at $[200, 400, 600, 800]$.

4.3. Evaluation

Quantitative Result For evaluation, we use LIVE-1 [31], testset of BSDS500 [3] and ICB [29] dataset. In the aspect of the JPEG decoder, we present the rate-distortion curve to illustrate the trade-off between bits-per-pixel (bpp) and peak signal-to-noise ratio (PSNR) where quality factors in a range of $[10, 100]$. We observed that BSDS500 [3] is saved as JPEG with a quality factor of 95. Therefore, the reported BSDS500 data is within a quality factor of 90. We compare our JDEC against existing compression artifact removal models: DnCNN [35], QGAC [14], and FBCNN [18] in Fig. 6. The selected models cover a relatively wide range of quality factors with a single network. We evaluate DnCNN [35] following the suggested method in QGAC [14], with channels being processed independently. Despite QGAC [14] having a training range of $[10, 100]$, it experiences a drop in performance in the range of $(90, 100]$ across all datasets. FBCNN [18] also exhibits a performance drop in the range of $[95, 100]$ when evaluated on the LIVE-1 [31] dataset. In comparison, JDEC outperforms all other methods, regardless of the quality factor or dataset.

	Luma($\mathbf{I_Y}$)|Chroma($\mathbf{I_C}$)			
Method	$q = 10$	$q = 20$	$q = 30$	$q = 40$
JPEG	34.39|35.77	37.32|38.90	39.85|40.72	39.82|39.59
DnCNN [35]	35.30|35.85	37.60|38.01	38.78|38.96	39.45|39.51
QGAC [14]	37.28|38.18	39.75|39.94	41.00|40.69	41.73|41.09
FBCNN [18]	37.12|38.36	39.71|40.21	40.97|41.04	41.81|41.50
JDEC (*ours*)	37.32|38.90	39.85|40.72	41.11|41.52	41.92|41.96

Table 2. Quantitative comparisons of each components in ICB [29] datasets. (PSNR(dB))

Regarding JPEG artifact removal, we report PSNR, structural similarity index (SSIM), and PSNR-B for estimating de-blocking in Tab. 1. We include DMCNN [36], IDCN [37], and transformer-based Swin2SR [9] as additional comparative groups since they cover a range of quality factors. Note that the Swin2SR has trained on a limited range of quality factors in a range of $[10, 40]$ with additional datasets, including the train and test dataset of BSDS500 [3] and Waterloo [26]. We partitioned the data presented in Tab. 1 to distinguish between networks operating within limited and expansive ranges. Our JDEC shows remarkable performance compared to other methods. The maximum PSNR interval is 0.37dB on ICB for $q = 10$.

We demonstrate Tab. 2 to observe the restoration effects

Test	LIVE-1 [31]			
Method	$q=80$	$q=90$	$q=95^*$	$q=100$
JPEG	34.23|33.45 0.948	36.86|36.45 0.967	39.33|38.90 0.979	43.07|42.37 0.993
DNCNN [35]	35.01|34.69 0.954	37.29|36.97 0.970	39.20|38.79 0.980	41.15|40.59 0.987
QGAC [14]	35.75|35.19 0.960	37.75|37.20 0.973	37.50| 37.01 0.974	38.97|38.56 0.979
FBCNN [18]	36.02|35.41 0.961	38.25|37.68 0.974	40.23|39.65 0.983	42.23|41.52 0.990
JDEC (*ours*)	36.31| 35.73 0.963	38.72|38.17 0.976	40.41|39.90 0.983	45.14|44.20 0.995
Test	ICB [29]			
Method	$q=80$	$q=90$	$q=95^*$	$q=100$
JPEG	36.34|35.82 0.891	37.72| 37.40 0.912	39.17|39.01 0.934	41.31|41.28 0.955
DNCNN [35]	35.57|35.44 0.844	36.75|36.64 0.868	37.99|37.92 0.891	39.73| 39.69 0.915
QGAC [14]	37.58|37.47 0.902	38.34|38.21 0.919	36.84|36.68 0.912	37.55|37.48 0.926
FBCNN [18]	38.03|37.91 0.902	39.17|39.03 0.920	40.36|40.22 0.938	41.61|41.52 0.951
JDEC (*ours*)	38.43|38.29 0.906	39.58|39.41 0.924	40.77|40.63 0.943	43.61|43.52 0.968

Table 3. Quantitative comparisons of high-quality images in LIVE-1 [31] and ICB [29] datasets (PSNR|PSNR-B(dB)) (top), SSIM (bottom). *: Quality factor 95 is a generally used default quality factor in the JPEG encoder.

ID	Method		Quality Factor q			
	g_ϕ-(a)	$\mathbf{T}_\psi$	10	20	30	40
0*	✓	✓	**27.95|27.71**	**30.26|29.87**	**31.59|31.12**	**32.50|31.98**
1	✓	✗	27.76|27.51	30.04|29.62	31.35|30.84	32.25|31.68
2	✗	✓	27.69|27.43	29.95|29.53	31.25|30.71	32.14|31.54
3	✗	✗	26.90|26.61	28.37|28.06	28.73|28.36	28.96|28.57
4	✓	Eq. (13)	27.88|27.64	30.21|29.83	31.54|31.07	32.45|31.93

Table 4. **Quantitative ablation study** of JDEC on LIVE-1 [31] (PSNR|PSNR-B (dB)). ∗: ID-0 is the proposed method JDEC. The definition of each ID number is shown in Sec. 4.4.

of two components of different sizes $\mathbf{I_Y} \in \mathbb{R}^{H\times W\times 1}, \mathbf{I_C} \in \mathbb{R}^{\frac{H}{2}\times\frac{W}{2}\times 2}$. According to Tab. 2, the performance difference in the chroma component $\mathbf{I_C}$ is greater than the difference of the luma component $\mathbf{I_Y}$ indicating an empirical upsampling effect.

In Tab. 3, we show the comparison of the high-quality image decoding. The * mark indicates the commonly used default quality factor of the JPEG, including OpenCV [6]. As a practical decoder for JPEG, only our JDEC decodes the best images among other baselines, including the conventional JPEG decoder.

Qualitative Result We show color JPEG artifact removal task in Fig. 7. There are two main distortions of JPEG compression: 1) lack of High-frequency components and 2) color differences. We demonstrate the effect of our JDEC in addressing the distortions in high frequencies in the first row of Fig. 7. While other methods suffer from aliasing, our JDEC successfully recovers the details of the butterfly's antennae. In the second row of Fig. 7, our JDEC relieves color distortion derived from JPEG compression.

In Fig. 8, we sort out the chroma components from each image to demonstrate the effect of our JDEC in relieving color distortions. When observing the chroma components using other methods, it is noticeable that the chroma components remain significantly distorted. However, our JDEC restores them closer to the original. It demonstrates that JDEC robustly restores color components subjected to quantization and downsampling, effectively mitigating distortion.

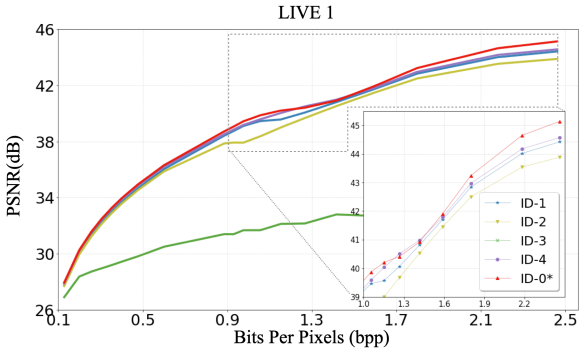


Figure 9. **Quantitative ablation study** of JDEC on LIVE-1 [31] (RD-curve), against ablation models. Our proposed JDEC achieves higher PSNR than any other models in most of the q values.

4.4. Ablation Study

Network Components We conducted ablation studies for the main components of our proposed JDEC. The proposed method, CCF $\mathbf{T}_\psi$ contains a frequency estimator which makes JDEC learn enhanced spectra. To support this, we train JDEC without a frequency estimator, directly forwarding concatenated coordinates (ID-1). We use additional 3×3 convolutional layers to have a comparable number of parameters. The sub-block conversion is the main element of encoder $\mathbf{E}_\varphi$. The spatial area gains a degree of freedom by using the sub-block conversion of the DCT matrix. We conduct the ablation study of sub-block conversion by embedding inputs directly (ID-2 of Tab. 4). The drop in performance is severe when both components are missing (ID-3 of Tab. 4). We also observed that ID-3, training without both the group embedding and CCF, leads to a significant performance drop as shown in Fig. 9.

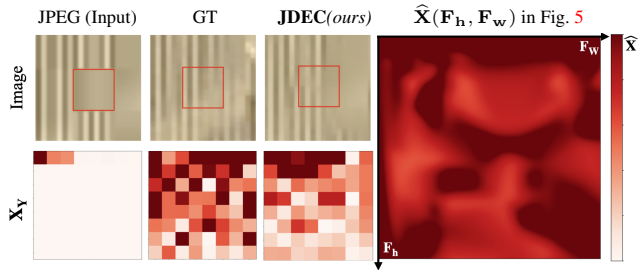
Fourier Features Comparing to the existing sinusoidal representation, the formulation of [20] will be compatible for our CCF. The modified Fourier feature is as follows :

$$\mathbf{C} \odot \begin{bmatrix} \cos(\pi(\mathbf{F}\cdot\delta + h_q(\mathbf{Q})) \\ \sin(\pi(\mathbf{F}\cdot\delta + h_q(\mathbf{Q})) \end{bmatrix}. \tag{13}$$

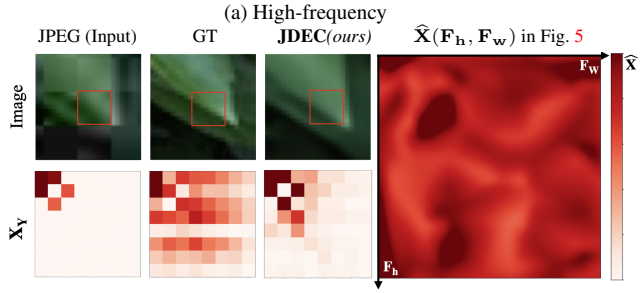
We label the model using Eq. (13) instead of $\mathbf{T}_\psi$ as ID-4. The rate-distortion curve of all ablation models is illustrated in Fig. 9. As shown in Fig. 9, the maximum gain of our CCF is 0.58dB against ID-4, where the quality factor is 100. Eq. (13) is considered as using additional terms than ID-0 (JDEC) by trigonometric sum. However, it has led to performance degradation as shown in Tab. 4.

4.5. Continuous Cosine Spectrum

In this section, we demonstrate that our CCF extracts dominant frequencies and amplitudes from highly compressed



(a) High-frequency



(b) Low-frequency

Figure 10. **Comparison of the estimated spectra of the Continuous Cosine Formulation (CCF).** The quality factor of input is 10. The estimated CCF spectrum follows the spectrum of ground-truth images despite severe distortion.

JPEG spectra. The ranges of input images (8×8) are highlighted with red boxes in each image. For visualization, we observe components of CCF, including estimated frequencies $\mathbf{F_h}, \mathbf{F_w}$ and amplitudes $\widehat{\mathbf{X}}$. We scatter frequencies in 2D space and assign a color to each amplitude. We quantize the frequencies to [0, 50] with steps of 1 and interpolate to continuous values. In Fig. 10a, most of the high-frequency components have been removed. The estimated spectrum with CCF is centered on high-frequency components despite such circumstances. In the case of Fig. 10b, the dominant components of the spectrum are focused on relatively low-frequency. Even in this case, the extracted spectrum of CCF is concentrated in the low-frequency elements as in the ground truth.

5. Discussion

Implicit Neural Representation As discussed in Sec. 2 and Eq. (4), a JPEG encoder downsamples chroma components. Therefore, the JPEG decoder should map: $\mathbb{R}^{\frac{H}{2}\times\frac{W}{2}\times 2} \mapsto \mathbb{R}^{H\times W\times 2}$ for chroma. Our JDEC addresses this issue through CCF (T_ψ) by embedding δ into 1×1-sized features, making the proposed JDEC a function of δ. Our model is able to decode high-resolution images when provided with dense coordinates that were not observed during training. We show the advanced additional upsampling results in the supplement material.

Extreme Reconstruction We primarily propose a decoding network to generate high-quality images due to its practical applicability. Consequently, we pursued the network without explicitly considering the scenario of high compression

Figure 11. **Reconstruction of the extremely compressed image** ($q = 0$) in LIVE-1 [31] dataset.

Method	#Params. (M)	Mem. (GB)	Time (ms)	FLOPs (G)	PSNR|PSNR-B (dB) $q=10$		PSNR|PSNR-B (dB) $q=40$	
FBCNN [18]	70.1	**0.61**	71.95	709.97	32.18	32.15	36.02	35.95
Swin2SR [9]	11.5≤	2.79	2203.59	3301.5	32.46	-	36.25	-
JDEC	38.9	1.76	224.79	1006.72	**32.55**	**32.51**	**36.37**	**36.28**
JDEC-CNN†	**26.2**	0.81	**56.59**	**476.33**	32.31	32.27	36.19	36.09

Table 5. Computational resources & performance comparison for a 560 × 560 pixels in ICB [29]. † :We replace the deep feature extractor of Fig. 4 with a CNN structure for comparison with the CNN-based model [18].

($q = 0$). However, by incorporating all image quality factors within the range [0,100] with a step size of 10 during the learning process, we successfully developed a decoding method tailored for highly compressed images. We label the additional network as JDEC+. As shown in Fig. 11, our JDEC+ recovers the highly compressed images better than image restoration models.

Computation Time and Memory In Tab. 5, we report computational resources including the number of parameters, memory consumption, floating-point operations (FLOPs), and computational time in GPU (NVIDIA RTX 3090 24GB). The input size is 560 × 560 for ours, while other methods have the size of 512 × 512.

6. Conclusion

We proposed a local implicit neural representation approach for decoding compressed color JPEG files. Our JPEG Decoder with Enhanced Continuous cosine coefficients (JDEC) contains a novel continuous cosine formulation (CCF) to extract a high-quality spectrum of images. JDEC takes a distorted spectrum as an input of the network and decodes it to a high-quality image regardless of the given quality factor. The suggested CCF extracts the dominant components of the ground truth spectrum, effectively. The results of benchmark datasets demonstrate that our network outperforms existing models as a practical JPEG decoder.

Acknowledgement This work was partly supported by Smart HealthCare Program (www.kipot.or.kr) funded by the Korean National Police Agency (KNPA) (No. 230222M01) and Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2021-0-02068, Artificial Intelligence Innovation Hub).

References

[1] Eirikur Agustsson and Radu Timofte. NTIRE 2017 Challenge on Single Image Super-Resolution: Dataset and Study. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2017. 5

[2] Nasir Ahmed, T. Natarajan, and Kamisetty R Rao. Discrete cosine transform. *IEEE transactions on Computers*, 100(1): 90–93, 1974. 2, 3

[3] Pablo Arbelaez, Michael Maire, Charless Fowlkes, and Jitendra Malik. Contour detection and hierarchical image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 33(5):898–916, 2010. 5, 6

[4] Yuval Bahat and Tomer Michaeli. What's in the image? explorable decoding of compressed images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2908–2917, 2021. 2

[5] Nuri Benbarka, Timon Höfer, Hamd ul-Moqeet Riaz, and Andreas Zell. Seeing Implicit Neural Representations As Fourier Series. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 2041–2050, 2022. 1

[6] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000. 5, 7

[7] Lukas Cavigelli, Pascal Hager, and Luca Benini. Cas-cnn: A deep convolutional neural network for image compression artifact suppression. In *2017 International Joint Conference on Neural Networks (IJCNN)*, pages 752–759. IEEE, 2017. 2

[8] Yunjin Chen and Thomas Pock. Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1256–1272, 2017. 2

[9] Marcos V Conde, Ui-Jin Choi, Maxime Burchi, and Radu Timofte. Swin2sr: Swinv2 transformer for compressed image super-resolution and restoration. In *European Conference on Computer Vision*, pages 669–687. Springer, 2022. 5, 6, 8

[10] Thomas M Cover. *Elements of information theory*. John Wiley & Sons, 1999. 1

[11] Chao Dong, Yubin Deng, Chen Change Loy, and Xiaoou Tang. Compression artifacts reduction by a deep convolutional network. In *Proceedings of the IEEE international conference on computer vision*, pages 576–584, 2015. 1, 2

[12] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Image super-resolution using deep convolutional networks. *IEEE transactions on pattern analysis and machine intelligence*, 38(2):295–307, 2015. 2

[13] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2020. 2

[14] Max Ehrlich, Larry Davis, Ser-Nam Lim, and Abhinav Shrivastava. Quantization guided jpeg artifact correction. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VIII 16*, pages 293–309. Springer, 2020. 1, 5, 6, 7

[15] Lionel Gueguen, Alex Sergeev, Ben Kadlec, Rosanne Liu, and Jason Yosinski. Faster neural networks straight from jpeg. *Advances in Neural Information Processing Systems*, 31, 2018. 1, 2

[16] W. Han, B. Lee, S. Park, and K. Jin. ABCD : Arbitrary bitwise coefficient for de-quantization. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5876–5885, 2023. 1, 4, 5

[17] Jianmin Jiang and Guocan Feng. The spatial relationship of dct coefficients between a block and its sub-blocks. *IEEE Transactions on Signal Processing*, 50(5):1160–1169, 2002. 4

[18] Jiaxi Jiang, Kai Zhang, and Radu Timofte. Towards flexible blind jpeg artifacts removal. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4997–5006, 2021. 1, 2, 5, 6, 7, 8

[19] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. 6

[20] Jaewon Lee and Kyong Hwan Jin. Local texture estimator for implicit representation function. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1929–1938, 2022. 1, 5, 7

[21] Jaewon Lee, Kwang Pyo Choi, and Kyong Hwan Jin. Learning local implicit fourier representation for image warping. In *European Conference on Computer Vision (ECCV)*, pages 182–200. Springer, 2022. 1, 4, 5

[22] Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. SwinIR: Image Restoration Using Swin Transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pages 1833–1844, 2021. 4

[23] Pengju Liu, Hongzhi Zhang, Kai Zhang, Liang Lin, and Wangmeng Zuo. Multi-level wavelet-cnn for image restoration. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 773–782, 2018. 2

[24] Pengju Liu, Hongzhi Zhang, Kai Zhang, Liang Lin, and Wangmeng Zuo. Multi-level wavelet-cnn for image restoration. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 773–782, 2018. 1

[25] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, Furu Wei, and Baining Guo. Swin transformer v2: Scaling up capacity and resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12009–12019, 2022. 2, 4, 6

[26] Kede Ma, Zhengfang Duanmu, Qingbo Wu, Zhou Wang, Hongwei Yong, Hongliang Li, and Lei Zhang. Waterloo Exploration Database: New challenges for image quality assessment models. *IEEE Transactions on Image Processing*, 26(2):1004–1016, 2017. 6

[27] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. NeRF:

Representing Scenes as Neural Radiance Fields for View Synthesis. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 1

[28] Jeongsoo Park and Justin Johnson. Rgb no more: Minimally-decoded jpeg vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22334–22346, 2023. 1, 2, 4

[29] Rawzor. Image compression benchmark. . *url: http://imagecompression.info/.* 5, 6, 7, 8

[30] C. E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27(3):379–423, 1948. 2

[31] Hamid R Sheikh, Muhammad F Sabir, and Alan C Bovik. A statistical evaluation of recent full reference image quality assessment algorithms. *IEEE Transactions on image processing*, 15(11):3440–3451, 2006. 5, 6, 7, 8

[32] Vincent Sitzmann, Julien Martel, Alexander Bergman, David Lindell, and Gordon Wetzstein. Implicit Neural Representations with Periodic Activation Functions. In *Advances in Neural Information Processing Systems*, pages 7462–7473. Curran Associates, Inc., 2020. 1

[33] G.K. Wallace. The jpeg still picture compression standard. *IEEE Transactions on Consumer Electronics*, 38(1):xviii–xxxiv, 1992. 1, 2, 5

[34] Kai Xu, Minghai Qin, Fei Sun, Yuhao Wang, Yen-Kuang Chen, and Fengbo Ren. Learning in the frequency domain. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1740–1749, 2020. 1, 2

[35] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE transactions on image processing*, 26(7):3142–3155, 2017. 1, 5, 6, 7

[36] Xiaoshuai Zhang, Wenhan Yang, Yueyu Hu, and Jiaying Liu. Dmcnn: Dual-domain multi-scale convolutional neural network for compression artifacts removal. In *2018 25th IEEE international conference on image processing (icip)*, pages 390–394. IEEE, 2018. 5, 6

[37] Bolun Zheng, Yaowu Chen, Xiang Tian, Fan Zhou, and Xuesong Liu. Implicit dual-domain convolutional network for robust color image compression artifact reduction. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(11):3982–3994, 2019. 1, 5, 6