Parameter-Efficient Harmonic Networks for JPEG Compression Artifact Removal

Hasan H. Karaoglu
Electronics and Communication Engineering Department
Istanbul Technical University
Istanbul, Türkiye
karaoglu16@itu.edu.tr

Ender M. Eksioglu
Electronics and Communication Engineering Department
Istanbul Technical University
Istanbul, Türkiye
eksioglue@itu.edu.tr

2023 International Conference on Machine Learning and Applications (ICMLA) | 979-8-3503-4534-6/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICMLA58977.2023.00310

***Abstract*—Many modern learning algorithms try to solve JPEG compression artifact removal (CAR) problem in pixel domain by mapping low-quality compressed image to high-quality image. Although JPEG artifacts arise from quantizing DCT coefficients of non-overlapped image blocks, researchers utilize transform domain as auxiliary information at most. On the other hand, it is well known and approved that extracting image blocks with overlap improves decompression performance. Inspired by these observations, we propose novel and fully transform domain convolutional neural networks (CNNs) for the problem. We choose DCT and DST, another effective DCT-like transform in terms of energy compactness, as the transform to be utilized and refer them as harmonic transforms. We perform harmonic transform on small overlapping blocks of compressed image in the first layer of the proposed networks, and we create spectral feature maps by properly ordering their harmonic transform coefficients with the same frequency. After a series of convolution blocks, we take inverse harmonic transform of the corresponding image blocks at the end of the network and put the resulting decompressed blocks back in their place. We show that forward and inverse transform layers of our harmonic networks are efficiently implemented with fast convolution and deconvolution layers by using 2D harmonic basis images as convolution kernels with a mathematical justification. Experimental study indicates that although our harmonic networks have a simple network topology and much fewer parameters than compared state-of-the-art deep networks, they are effective and efficient to suppress compression artifacts and give comparable results.**

***Index Terms*—JPEG compression artifact removal, deep learning, discrete cosine transform, discrete sine transform**

I. Introduction

Digital images are compressed for reasons such as less memory and faster transmission. JPEG [1] is one of the most popular modern image compression methods. Typical steps of JPEG compression scheme are to split an image into nonoverlapping 8×8 image blocks, to perform 2D forward DCT on them, to quantize the resulting block DCT spectrums with a suitable quantization table, to perform inverse block DCT on the quantized spectral coefficients, and to tile the compressed blocks. We note that better energy compaction property of the DCT compared to the other signal transforms is the reason for its use in the JPEG algorithm. Since compression process yields artifacts such as blockiness, ringing, and banding due to the quantization and block discontinuities [2], achieving a high quality image from the compressed image is a crucial task for nice photographic images and computer vision applications. However, CAR is an ill-posed problem [2] meaning that obtaining high-quality image is not a unique process.

Existing CAR algorithms in the literature can be categorized into three parts: model-based, learning-based, and hybrid algorithms. Although early model-based techniques relying on filtering [3] are simple and efficient, they give poor images with blurring artifacts. Regularization techniques, subsequent model-based algorithms, try to solve the problem by imposing some structural information (i.e., regularizer) such as non-local self-similarity [4], low-rankness [5], and transform sparsity [6], [7] on the compressed image. The downsides of regularization techniques are that (1) the lack of one global regularizer for photographic images containing many complex patterns such as textures, flat areas, and edges etc. and (2) the need of high computation cost due to the hard optimization procedure. ARCNN [8], TNRD [9], DnCNN [10], and MemNet [11] are representative successful (deep) learning based CAR algorithms. However, they tackle the problem by seeking a map from compressed image to ground truth image in spatial domain. Despite the source of JPEG artifacts originates from the nonlinear mapping of transform coefficients, i.e., quantization, the established practice is to utilize them as auxiliary information [12], [13]. On the other hand, hybrid approaches such as deep plug-and-play (PnP) CNN priors [14] and algorithm unrolling [15] target to combine the virtues of model- and learning-based CAR methods via Gaussian denoiser networks and iterative optimization algorithms, respectively. However, their drawbacks are that while PnP CNN priors require strong pretrained Gaussian networks with all noise levels, unrolled networks demand high computational resources for more iterations.

As noted earlier, although compression artifacts originate from transform domain, there are few approaches that solve the problem with deep networks in the transform domain. Some of them such as [16], [17] choose wavelet transform as the transform of interest since it provides subband images as transform coefficients which are suitable for convolution layers to seek a correlation between neighbor pixels. The other transform-

This work was supported by ITU BAP (Istanbul Technical University Research Fund) under project number 42027 (MDK-2019-42027).

2023 International Conference on Machine Learning and Applications (ICMLA) | 979-8-3503-4534-6/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICMLA58977.2023.00310

based deep techniques utilize DCT on small nonoverlapping image blocks [12], [13]. However, such treatment does not make use of the correlation of pixels on block boundaries. Whereas in smooth regions, for example, it is reasonable and effective to use pixels at block boundaries. It is also known from the signal processing literature that processing with overlapping blocks significantly improves an algorithm's decompression performance. In the light of these facts, we propose DCT and DST domain two networks to tackle JPEG artifacts.

The main contributions of this work are two-fold. (1) We design two novel and fully transform domain, i.e., DCT and DST domain, CNNs with a basic topology for reducing JPEG compression artifacts. When designing our networks, we make use of local DCT and DST spectra. We show that the DCT and DST spectra of overlapping image blocks can be arranged in such a way that the coefficients with the same spectral component form a channel, and this can be quickly implemented with fast convolution layer on GPU, which we refer them to harmonic filterbanks. We propose a deep CNN as a nonlinearity function and train them in an end-to-end manner. (2) Experimental study is conducted to show the effectiveness and efficiency of our networks by giving quantitative and qualitative results.

II. Background

A. DCT & DST

The DCT and DST [18] [1] are Fourier-related signal transforms and their aim is to separate a signal into harmonic cosine and sine basis vectors, respectively. Unlike the DFT [18], both of the transforms generate real transform coefficients and have been widely used in many practical applications especially signal denoising and compression due to the energy compaction property, which collect most of the signal energy on a few of its harmonic transform coefficients [18].

The definition of a 2D forward transform of an image $x(k,l)$ for $0 \leq k \leq M-1$ and $0 \leq l \leq N-1$ can be written by

$$\widetilde{X}(m,n) = \sum_{l=0}^{N-1} \sum_{k=0}^{M-1} x(k,l)\, w(k,l,m,n), \qquad (1)$$

where the elements of $\widetilde{X}(m,n)$ are called 2D forward transform coefficients for the frequency indices $0 \leq m \leq M-1$ and $0 \leq n \leq N-1$. The term $w(k,l,m,n)$ in (1) is called transformation kernel and can be picked depending on the transform utilized. In the literature, the two harmonic transforms have belong to eight transform definitions for different symmetry and boundary conditions [18]. The most popular DCT definition is the type-II DCT whose 1D DCT kernel $w_{DCT}(k,m)$ is defined as

$$w_{DCT}(k,m) = \alpha_k \cos\left[\frac{(2k+1)\, m\pi}{2M}\right], \qquad (2)$$

[1] In this work, we refer the DCT and DST to harmonic transforms.

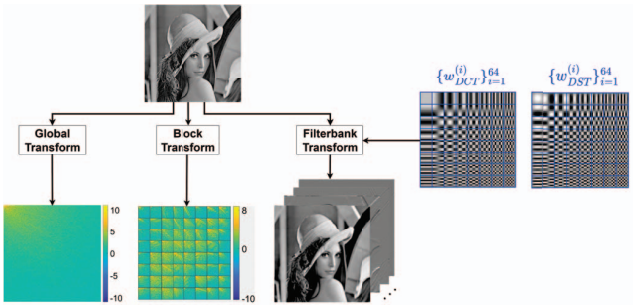

Fig. 1. Three DCT magnitude spectrums of Lena. Black and blue lines for block spectrum and DCT/ DST basis images are inserted for visualization. Due to the high dynamic range, the global and block spectrums are logarithmically scaled and rendered with colors to easily observe the large coefficients as opposed to the filterbank spectrum.

with the coefficients $\alpha_k = \sqrt{\frac{1}{M}}$ for $k = 0$ and $\alpha_k = \sqrt{\frac{2}{M}}$ for $1 \leq k \leq M-1$. The 1D kernel $w_{DST}(k,m)$ for the most commonly used DST is defined as

$$w_{DST}(k,m) = \sqrt{\frac{2}{M+1}} \sin\left[\frac{(k+1)(m+1)\pi}{M+1}\right]. \qquad (3)$$

Since 2D harmonic transforms are implemented in a seperable way [18], the 1D transform kernels in (2) and (3) can be easily extended to the 2D case. The DCT and DST basis images of size 8×8 are shown in Fig.

Despite of their similiar properties, there exist some differences between the two harmonic transform. The assumption of the DCT on a 1D N-point signal is $2N$-point periodic and even symmetric. However, the DST assumes that the signal is $(2N+1)$-point periodic and odd symmetric. As a result, only the DCT has a DC basis, not the DST. The importance of the transforms with DC components lies in keeping the average energy of the signal.

In the literature, there exist four types of DCT usages which can be extended to the DST case. *Global DCT* calculates the DCT spectrum of a whole image. *Block (local) DCT* which is utilized in JPEG compression scheme splits an image into small nonoverlapping blocks, takes 2D forward DCT on all of the blocks, and then tile the DCT coefficients in the same order. In *sliding window DCT* [19], block transform is performed on overlapping blocks. *DCT filterbank (FB)* utilizes DCT basis images as a convolution kernel [20]. Fig. 1 shows the global, block, and filterbank spectrums of Lena.

B. Harmonic Filterbanks

Transform-based CAR algorithms on overlapping image blocks can be written by

$$\widehat{\boldsymbol{x}}_k = \boldsymbol{W}^T \Upsilon\left(\boldsymbol{W}\boldsymbol{y}_k\right), \qquad (4)$$

where $\boldsymbol{y}_k = \boldsymbol{R}_k\boldsymbol{y} \in \mathbb{R}^n$ and $\widehat{\boldsymbol{x}}_k = \boldsymbol{R}_k\widehat{\boldsymbol{x}} \in \mathbb{R}^n$ are the kth image blocks extracted from the compressed image $\boldsymbol{y} \in \mathbb{R}^N$ and restored image $\widehat{\boldsymbol{x}} \in \mathbb{R}^N$, respectively. We note that n and N represents block and image sizes. In this work, we remain loyal to the notational convention in image restoration

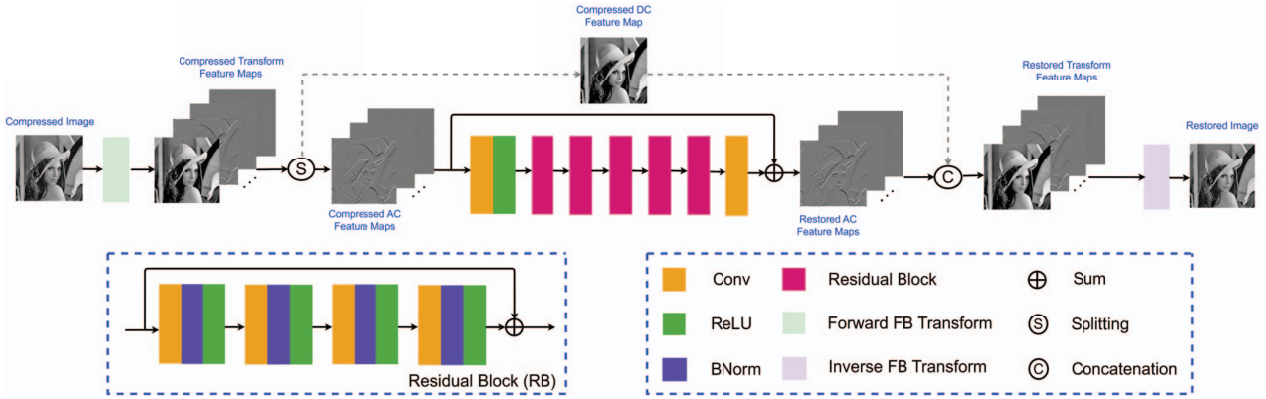

Fig. 2. Harmonic network architectures (DCTNet and DSTNet). Splitting and concatenation layers (dashed line) are only available for DCTNet.

literature. That is, we show an image as a vector which is stacked in a column-wise. The matrix $\boldsymbol{W} \in \mathbb{R}^{n\times n}$ is a harmonic transform matrix each row $\boldsymbol{w}_r^T$ of which coincides a basis vector and $\boldsymbol{W}^T$ is the transpose of $\boldsymbol{W}$. The function $\Upsilon(.)$ represents a fixed or learnable nonlinear function. The image block extraction matrix $\boldsymbol{R}_k \in \mathbb{R}^{n\times N}$ whose entries contain only zero and one terms extracts the kth image block from the corresponding image. Assuming that the image $\boldsymbol{y}$ is padded circularly, the decompressed image $\widehat{\boldsymbol{x}}$ is obtained by aggregating and averaging all of the restored blocks as follows:

$$\widehat{\boldsymbol{x}} = \frac{1}{n}\sum_{i=1}^{N} \boldsymbol{R}_i^T \boldsymbol{W}^T \Upsilon(\boldsymbol{W}\boldsymbol{R}_i\boldsymbol{y}). \tag{5}$$

If we write $\boldsymbol{W}_k = \boldsymbol{W}\boldsymbol{R}_k \in \mathbb{R}^{n\times N}$, the image $\widehat{\boldsymbol{x}}$ can be represented in a more compact form as follows:

$$\widehat{\boldsymbol{x}} = \boldsymbol{H}^T \Upsilon\left(\boldsymbol{H}\boldsymbol{y}\right). \tag{6}$$

Here, $\boldsymbol{H} = (1/\sqrt{n})\left[\boldsymbol{W}_1^T\ \boldsymbol{W}_2^T\ \dots\ \boldsymbol{W}_N^T\right]^T \in \mathbb{R}^{nN\times N}$. The harmonic transform spectrum $\boldsymbol{H}\boldsymbol{y}$ contains magnitudes of different frequencies to synthesize the image of interest. Since global and block transform coefficients of highly correlated signals such as images tend to be uncorrelated [21] as shown in Fig. 1, seeking for correlation between the spectrum coefficients and performing convolution on them is an ineffective attempt. Instead, as pointed out in [20], [22], packing the coefficients with the same frequency as a channel provides us feature maps which are suitable to be processed by convolution layers. This ordering scheme only permutes rather than unchanges the spectrum coefficients. The formal way of this spectral permutation step is to multiply $\boldsymbol{H}\boldsymbol{y}$ with a suitable permutation matrix $\boldsymbol{P} \in \mathbb{R}^{nN\times nN}$ as follows:

$$\widehat{\boldsymbol{x}} = \boldsymbol{H}^T\boldsymbol{P}^T \Upsilon\left(\boldsymbol{P}\boldsymbol{H}\boldsymbol{y}\right) = \boldsymbol{S}^T \Upsilon\left(\boldsymbol{S}\boldsymbol{y}\right), \tag{7}$$

where $\boldsymbol{S} \in \mathbb{R}^{nN\times N}$ and $\boldsymbol{S}^T \in \mathbb{R}^{N\times nN}$ are called forward and inverse harmonic (FB) transforms. The inner term $\boldsymbol{S}\boldsymbol{y}$ in (7) can be explicitly written by $\widetilde{\boldsymbol{y}} = \boldsymbol{S}\boldsymbol{y} = \left[\widetilde{\boldsymbol{y}}_1^T\ \widetilde{\boldsymbol{y}}_2^T\ \dots\ \widetilde{\boldsymbol{y}}_n^T\right]^T$. Each $\widetilde{\boldsymbol{y}}_r \in \mathbb{R}^N$ for $1 \le r \le n$ is called a subband image. As justified in [20], [22], the rth subband image $\widetilde{\boldsymbol{y}}_r$ can be obtained by convolving the rth harmonic basis $\boldsymbol{w}_r^T$ with the compressed image $\boldsymbol{y}$ for circular padding condition as follows:

$$\widetilde{\boldsymbol{y}}_r = \frac{1}{\sqrt{n}}\left(\boldsymbol{w}_r^T \otimes \boldsymbol{y}\right). \tag{8}$$

Here, $\otimes$ denotes convolution operation. Similarly, the inverse transform can be efficiently implemented with a deconvolution layer with the same basis images as follows:

$$\widehat{\boldsymbol{x}} = \frac{1}{\sqrt{n}}\sum_{r=1}^{n} \boldsymbol{w}_r \otimes \Upsilon(\widetilde{\boldsymbol{y}}). \tag{9}$$

If we choose DCT basis images for harmonic FB transform, the first subband image $\widetilde{\boldsymbol{y}}_1$ is called DC subband image and is denoted by $\widetilde{\boldsymbol{y}}_{DC}$. The remaining subband images $\widetilde{\boldsymbol{y}}_r$ for $2 \le r \le n$ are called AC subband images. When we pick DST basis images for the FB usage, we have no DC subband image due to the lack of DC basis image of the DST. Hence, DST FB generates only AC subband images.

III. Harmonic Networks - DCTNet & DSTNet

Building harmonic networks' architecture requires to determine two design choices. The first criteria is to choose 2D basis images for the harmonic transform. As depicted earlier, we have two transform options as the DCT and DST. The other design criteria is to determine the nonlinearity function $\Upsilon(.)$. The literature on image restoration problems, including CAR, contains many studies on the selection of the function. Fixed-shape scalar functions $\Upsilon : \mathbb{R} \to \mathbb{R}$ with tunable parameter(s) to operate on the whole or each subband image such as soft and hard thresholding have been proposed for image denoising problem. However, since determining the shape of the function in advance directly affects the performance, Hel-Or and Ben-Artzi [20] proposes learning the shape of each function by modeling with a sum of some special functions. Maharjan et al. [23] proposes a CNN as the nonlinearity function for each band obtained by block DCT. In doing so, it increases the computational burden of the algorithm and does not take advantage of the correlation between pixels at adjacent block

boundaries produced by block DCT. On the other hand, there is a clear relationship between neighboring spectral coefficients in each band, but this case is not taken into account in deep networks. Keeping in mind that CNNs are powerful function approximators and inspired by [22], we propose any deep CNN as the nonlinearity function: We try to keep the architecture of our networks simple and efficient, and for this purpose, we make use of residual blocks. This is because residual blocks facilitate the training of deep networks and preserve the training stability. Apart from this, our networks have simple CNN structures compared to the state-of-the-art deblocking networks.

The architecture of the proposed networks is visualized in Fig. 2. It is worth noting that the main distinction between the two harmonic networks is the lack of splitting and concatenation layers for the DSTNet since the DST yields no average energy. Hence, DSTNet has more learnable parameters than DCTNet's since all the subband images of DSTNet are given to a series of residual blocks. The forward FB transform block takes the compressed image $\boldsymbol{y}$ and produces the subband images $\widetilde{\boldsymbol{y}}$. The DC subband is left untouched because of the importance of conserving average energy in signal and image processing applications (only valid for DCTNet). The AC feature maps are given to one convolution (Conv) + rectifier linear unit (ReLU) block followed by 5 residual block (RB) and one Conv layer. Each RB contains 4 Conv + BN + ReLU layers and one sum connection, which BN is batch normalization in short. The feature maps yielded by the last Conv layer is concatenated with the DC feature map $\widetilde{\boldsymbol{y}}_{DC}$ (only valid for DCTNet). At the last step, the resulting feature maps are mapped to image domain by inverse FB transform block.

Assuming that we have N compressed and ground truth training pairs $\{\boldsymbol{y}^{(i)}, \boldsymbol{x}^{(i)}\}_{i=1}^{N}$, the loss function of the proposed networks to train in supervised learning is written by

$$\mathcal{L}(\boldsymbol{\Theta}) = \frac{1}{2N} \sum_{i=1}^{N} \|\boldsymbol{x}^{(i)} - \mathcal{D}(\boldsymbol{y}^{(i)}; \boldsymbol{\Theta})\|_2^2. \tag{10}$$

Here, $\mathcal{L}(.)$ is the loss function to be optimized. $\mathcal{D}(.)$ and $\boldsymbol{\Theta}$ denote our harmonic network and its learnable parameters.

IV. Experiments

We train four networks with four quality factors (QFs), i.e., Q = 10, 20, 30, and 40 for each harmonic network for reducing JPEG compression artifacts on grayscale images. BSDS500 [24] is used to create a large compressed and uncompressed training patches. 133K uncompressed patches of size 60×60 is extracted from 400 training images to make more use of the dataset. During training stage, data augmentation steps such as rotation and flipping have also been adopted. JPEG low-quality compressed images for the QFs given above are generated by the MATLAB JPEG encoder. The training and testing phases of the proposed networks are conducted on MatConvNet [25] deep learning framework which is built upon Matlab (2019a) on a desktop computer with an Intel Core i7-8700k CPU 3.2 GHz, 64-bit operating system, 16GB memory, and a Nvidia GeForce RTX2080 Ti GPU. The loss function to train our networks in (10) is optimized via Adam [26] with a mini-batch size of 64. All learnable parameters are initialized with He initialization scheme. The learning rate is scheduled from $1e-2$ to $1e-5$ with an exponential decaying scheme. The weight decay parameter is set to $1e-4$. The forward and inverse FB transform layers of the proposed networks use a total of 49 basis images of size 7×7 and all of the remaining convolution layers use 49 filters of size $3 \times 3 \times 49$ and 48 filters of size $3 \times 3 \times 48$ for DCTNet and DSTNet, respectively. The training time of our networks takes 34 hours with the hardware whose specifications are given above.

We evaluate our harmonic networks on two benchmark datasets, namely, Classic5 (5 test images) [7] and LIVE1 (24 test images) [29]. The test sets are not included into the training datasets. We use the publicly avaliable codes for all of the compared methods. As quantitative performance metrics, the Peak Signal-to-Noise Ratio (PSNR, measured in dB) and the Structural Similarity Index (SSIM) are selected. Algorithms selected for comparison are ARCNN [8], TNRD [9], DnCNN [10], MemNet [11], QGAC [27], IACNN [28], and DUN [15]. ARCNN [8] is the seminal work whose architecture is a three-layer vanilla CNN. TNRD [9] is the first unrolled neural network with a sum of radial basis functions. DnCNN [10] is another CNN algorithm using batch normalization and residual learning paradigms for the first time. MemNet [11] uses long- and short-term memory connections and gate mechanisms to attack the problem. The architecture of QGAC [27] is built on several sophisticated blocks such as frequencyNet, blockNet, and fusion network. IACNN [28], deep CNN with two inception-blocks, proposes a deep classification network for estimating the quality factor Q. DUN [15] is a hybrid method that unfolds the iterative algorithm by modeling JPEG residuals with a convolutional dictionary.

V. Results

The average PSNR and SSIM results on the two test sets are reported in Table I for the four QFs. In the same table, the total number of learnable parameters and the parameter gains in percentage terms compared to the number of parameters of DCTNet are also given. As seen from the table, while our two networks surpass ARCNN and TNRD dramatically at the cost of increased parameter number in terms of average PSNR, DCTNet and DSTNet have an average performance gain of 0.26 and 0.24 dB over DnCNN, despite having 28.55% fewer parameters than DnCNN. While the proposed networks have 28.34% fewer parameters than MemNet, they show comparable performance results to MemNet which can be attributed that MemNet has short- and long-term memories, recursive gate units and multi-supervision scheme. In terms of SSIM index, the performance of our networks is better than the PSNR results. It can be readily inferred that our DCTNet and DSTNet show similar performance results for the two test sets. Our harmonic networks beat IACNN by 0.2 dB on average, with about 4 million fewer parameters. Despite having 96% fewer parameters than QGAC, a more sophisticated network,

TABLE I

THE AVERAGE PSNR(DB) AND SSIM RESULTS OF DIFFERENT METHODS ON CLASSICS5 AND LIVE1 DATASETS. THE BEST RESULTS ARE HIGHLIGHTED IN BOLD. NEGATIVE GAIN DENOTES THAT COMPARED NETWORK HAS FEWER PARAMATERS THAN BASELINE DCTNET.

Datasets	Q	JPEG	ARCNN [8]	TNRD [9]	DnCNN [10]	MemNet [11]
Classics5	10	27.82 / 0.7595	29.03 / 0.7929	29.28 / 0.7992	29.40 / 0.8026	29.69 / 0.8107
	20	30.12 / 0.8344	31.15 / 0.8517	31.47 / 0.8576	31.63 / 0.8610	31.90 / 0.8658
	30	31.48 / 0.8666	32.51 / 0.8806	32.78 / 0.8837	32.91 / 0.8861	-
	40	32.43 / 0.8849	32.68 / 0.9019	-	33.77 / 0.9141	-
LIVE1	10	27.77 / 0.7730	28.96 / 0.8076	29.15 / 0.8111	29.19 / 0.8123	29.45 / 0.8193
	20	30.07 / 0.8512	31.29 / 0.8733	31.46 / 0.8769	31.59 / 0.8802	31.83 / 0.8846
	30	31.40 / 0.8851	32.67 / 0.9043	32.84 / 0.9059	32.98 / 0.9090	-
	40	32.35 / 0.9041	32.74 / 0.9196	-	33.96 / 0.9346	-
# Params. / Gain	-	-	106K / -350.94%	26K / -1738.46%	669K / 28.55%	667K / 28.34%
Datasets	Q	QGAC [27]	IACNN [28]	DUN [15]	DCTNet	DSTNet
Classics5	10	29.84 / **0.8370**	29.53 / 0.8124	**29.95** / 0.8343	29.67 / 0.8109	29.64 / 0.8102
	20	31.98 / **0.8850**	31.87 / 0.8729	**32.11** / 0.8848	31.89 / 0.8659	31.84 / 0.8649
	30	33.22 / **0.9070**	33.08 / 0.9007	**33.33** / 0.9061	33.15 / 0.8899	33.16 / 0.8896
	40	-	33.91 / 0.9141	**34.11** / 0.9179	34.02 / 0.9037	33.99 / 0.9030
LIVE1	10	29.53 / **0.8400**	28.80 / 0.8207	**29.61** / 0.8370	29.43 / 0.8214	29.45 / 0.8209
	20	31.86 / **0.9010**	31.76 / 0.8861	**31.98** / 0.8997	31.83 / 0.8862	31.81 / 0.8850
	30	33.23 / 0.9250	33.14 / 0.9210	**33.38** / **0.9251**	33.26 / 0.9139	33.25 / 0.9131
	40	-	34.06 / 0.9313	**34.32** / 0.9384	34.25 / 0.9290	34.24 / 0.9282
# Params. / Gain	-	12000K / 96.02%	4321K / 88.94%	10490K/ 95.44%	478K / 0%	479K / 0 %

Fig. 3. Comparative visual results of the compressed image Barbara with $Q = 20$. Please zoomed-in view for better visualization. (a) Clean, PSNR / SSIM. (b) JPEG, 28.34/ 0.8535. (c) DnCNN [10], 30.57 / 0.8913. (d) DCTNet, 31.20 / 0.8999. (e) DSTNet, 31.06 / 0.8983.

the proposed networks are about 0.1 dB behind QGAC. As a final comparison, our networks are outperformed the hybrid unfolded network DUN by about 0.25 dB on average for 10 million fewer learnable parameter gains. All these comparisons show that our networks are more suitable for parameter-efficient platforms such as mobile devices. Fig. 3 gives the visual results for the test image Barbara for $Q = 20$. We note that proposed DCTNet performs better than DnCNN for recovering the textural details. The second best result belongs to the DSTNet which beats DnCNN with a 0.15 dB PSNR difference.

In order to show the efficiency and robustness of the proposed methods, we also conduct additional experiments examining the impact of varying the number of 2D harmonic basis images on deblocking performance. We train our networks for 5×5 and 3×3 2D harmonic basis images. We note that the networks with 7×7 corresponds to our original harmonic networks. The total number of learnable parameters for the harmonic networks is directly dependent on the number of harmonic basis images. The PSNR and SSIM results of this ablation study are listed in Table II. In all 5×5 networks, the performance loss compared to the original harmonic networks was 0.1 dB, while the performance loss was 0.3 dB in the 3x3 DCTNet and 0.4 dB in the 3x3 DSTNet. However, It is noteworthy that the 3x3 DCTNet outperforms TNRD and ARCNN despite having far fewer parameters.

VI. CONCLUSION

In this paper, we propose DCT and DST based networks for the compression artifact removal problem. First, we take overlapping image blocks from the compressed image and arrange their harmonic transform coefficients in such a way that the coefficients with the same frequency form subband images. When returning to the pixel domain, we take the inverse transform of the overlapping blocks to form the whole image. We show that these transformations can actually be performed quickly with GPU-accelerated convolution and deconvolution layers using harmonic transform basis images for a unit stride. Proposed architectures utilize deep CNNs instead of a scalar functions with predetermined shape as the nonlinear function. The experimental results of the proposed networks trained with supervised learning verify that despite having much fewer parameters compared to the state-of-the-art deep networks, they give comparable performance to suppress compression artifacts. The quantitative and qualitative results also validate the efficiency of the proposed networks.

TABLE II
COMPARISON OF HARMONIC NETWORK MODELS WITH DIFFERENT NUMBER OF 2B HARMONIC BASIS IMAGES FOR $Q = 20$. PARAMETERS DENOTES TOTAL NUMBER OF MODEL PARAMETERS. TEST SETS ARE CLASSICS5 AND LIVE1.

Network	Size	# Params.	Dataset	PSNR / SSIM
DCTNet	3×3	16K	Classics5	31.50 / 0.8591
	5×5	125K	Classics5	31.79 / 0.8643
	7×7	478K	Classics5	31.89 / 0.8659
	3×3	16K	LIVE1	31.50 / 0.8803
	5×5	125K	LIVE1	31.74 / 0.8847
	7×7	478K	LIVE1	31.83 / 0.8862
DSTNet	3×3	16K	Classics5	31.43 / 0.8574
	5×5	125K	Classics5	31.77/ 0.8638
	7×7	479K	Classics5	31.84 / 0.8649
	3×3	16K	LIVE1	31.47 / 0.8785
	5×5	125K	LIVE1	31.74 / 0.8840
	7×7	479K	LIVE1	31.81 / 0.8850

REFERENCES

[1] Gregory K Wallace, "The JPEG Still Picture Compression Standard," *IEEE Transactions on Consumer Electronics*, vol. 38, no. 1, pp. 18–34, 1992.

[2] Jiaying Liu, Dong Liu, Wenhan Yang, Sifeng Xia, Xiaoshuai Zhang, and Yuanying Dai, "A Comprehensive Benchmark for Single Image Compression Artifact Reduction," *IEEE Transactions on Image Processing*, vol. 29, pp. 7845–7860, 2020.

[3] Shigenobu Minami and Avideh Zakhor, "An Optimization Approach for Removing Blocking Effects in Transform Coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 5, no. 2, pp. 74–82, 1995.

[4] Xinfeng Zhang, Ruiqin Xiong, Siwei Ma, and Wen Gao, "Reducing Blocking Artifacts in Compressed Images via Transform-domain Nonlocal Coefficients Estimation," in *IEEE International Conference on Multimedia and Expo*, 2012, pp. 836–841.

[5] Jie Ren, Jiaying Liu, Mading Li, Wei Bai, and Zongming Guo, "Image Blocking Artifacts Reduction via Patch Clustering and Low-rank Minimization," in *IEEE Data Compression Conference*, 2013, pp. 516–516.

[6] Xianming Liu, Xiaolin Wu, Jiantao Zhou, and Debin Zhao, "Data-driven Sparsity-based Restoration of JPEG-compressed Images in Dual Transform-pixel Domain," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5171–5178.

[7] Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian, "Pointwise Shape-adaptive DCT for High-quality Denoising and Deblocking of Grayscale and Color Images," *IEEE Transactions on Image Processing*, vol. 16, no. 5, pp. 1395–1411, 2007.

[8] 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*, 2015, pp. 576–584.

[9] 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*, vol. 39, no. 6, pp. 1256–1272, 2016.

[10] Kai Zhang, W. 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*, vol. 26, no. 7, pp. 3142–3155, 2017.

[11] Ying Tai, Jian Yang, Xiaoming Liu, and Chunyan Xu, "MemNet: A Persistent Memory Network for Image Restoration," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 4539–4547.

[12] Xiaoshuai Zhang, Wenhan Yang, Yueyu Hu, and Jiaying Liu, "DMCNN: Dual-domain Multi-scale Convolutional Neural Network for Compression Artifacts Removal," in *25th IEEE International Conference on Image Processing (ICIP)*, 2018, pp. 390–394.

[13] 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*, vol. 30, no. 11, pp. 3982–3994, 2019.

[14] Kai Zhang, Yawei Li, W. Zuo, Lei Zhang, Luc Van Gool, and Radu Timofte, "Plug-and-play Image Restoration with Deep Denoiser Prior," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 10, pp. 6360–6376, 2021.

[15] Xueyang Fu, Menglu Wang, Xiangyong Cao, Xinghao Ding, and Zheng-Jun Zha, "A Model-Driven Deep Unfolding Method for JPEG Artifacts Removal," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 11, pp. 6802–6816, 2022.

[16] 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*, 2018, pp. 773–782.

[17] Honggang Chen, Xiaohai He, Linbo Qing, Shuhua Xiong, and Truong Q Nguyen, "DPW-SDNet: Dual pixel-wavelet domain deep CNNs for soft decoding of JPEG-compressed images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 711–720.

[18] Rafael C Gonzalez and Richard E Woods, *Digital Image Processing*, Prentice Hall, Upper Saddle River, N.J., 2008.

[19] Leonid P Yaroslavsky, "Local Adaptive Image Restoration and Enhancement with the Use of DFT and DCT in a Running Window," in *Wavelet Applications in Signal and Image Processing IV*. SPIE, 1996, vol. 2825, pp. 2–13.

[20] Yacov Hel-Or and Gil Ben-Artzi, "The Role of Redundant Bases and Shrinkage Functions in Image Denoising," *IEEE Transactions on Image Processing*, vol. 30, pp. 3778–3792, 2021.

[21] Anil K Jain, *Fundamentals of Digital Image Processing*, Prentice-Hall, Inc., 1989.

[22] Hasan H. Karaoglu and Ender M. Eksioglu, "DCTNet: Deep Shrinkage Denoising via DCT Filterbanks," *Signal, Image and Video Processing*, vol. 17, no. 7, pp. 3665–3676, 2023.

[23] Paras Maharjan, Ning Xu, Xuan Xu, Yuyan Song, and Zhu Li, "DCTResNet: Transform Domain Image Deblocking for Motion Blur Images," in *2021 International Conference on Visual Communications and Image Processing*, 2021, pp. 1–5.

[24] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A Database of Human Segmented Natural Images and its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics," in *Proceedings 8th IEEE International Conference on Computer Vision*, 2001, vol. 2, pp. 416–423.

[25] Andrea Vedaldi and Karel Lenc, "MatConvNet: Convolutional Neural Networks for Matlab," in *Proceedings of the 23rd ACM International Conference on Multimedia*, 2015, pp. 689–692.

[26] Diederik P. Kingma and Jimmy Ba, "Adam: A Method for Stochastic Optimization," in *3rd International Conference on Learning Representations, San Diego, CA, USA*, 2015.

[27] Max Ehrlich, Larry Davis, Ser-Nam Lim, and Abhinav Shrivastava, "Quantization Guided JPEG Artifact Correction," in *16th European Conference on Computer Vision*. Springer, 2020, pp. 293–309.

[28] Yoonsik Kim, Jae Woong Soh, Jaewoo Park, Byeongyong Ahn, Hyun-Seung Lee, Young-Su Moon, and Nam Ik Cho, "A Pseudo-blind Convolutional Neural Network for the Reduction of Compression Artifacts," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 4, pp. 1121–1135, 2019.

[29] H Sheikh, "Live image quality assessment database release 2," *http://live. ece. utexas. edu/research/quality*, 2005.