Dual-Domain Learning for JPEG Artifacts Removal

Guang Yang, Lu Lin, Chen Wu and Feng Wang

November 7, 2023

Dual-Domain Learning For JPEG Artifacts Removal

Guang Yang[1][0009−0004−4899−3248], Lu Lin[3][0009−0003−0589−6694], Chen Wu[1]✉[0009−0002−1740−0804], and Feng Wang[2][0000−0001−8408−0527]

[1] University of Science and Technology of China, China
[2] Hefei Institute of Physical Science, Chinese Academy of Sciences, China
[3] Central China Normal University, China
{guangyang,wuchen5X}@mail.ustc.edu.cn

Abstract. JPEG compression brings artifacts into the compressed image, which not only degrades visual quality but also affects the performance of other image processing tasks. Many learning-based compression artifacts removal methods have been developed to address this issue in recent years, with remarkable success. However, existing learning-based methods generally only exploit spatial information and lack exploration of frequency domain information. Exploring frequency domain information is critical because JPEG compression is actually performed in the frequency domain using the Discrete Cosine Transform (DCT). To effectively leverage information from both the spatial and frequency domains, we propose a novel Dual-Domain Learning Network for JPEG artifacts removal (D2LNet). Our approach first transforms the spatial domain image to the frequency domain by the fast Fourier transform (FFT). We then introduce two core modules, Amplitude Correction Module (ACM) and Phase Correction Module (PCM), which facilitate interactive learning of spatial and frequency domain information. Extensive experimental results performed on color and grayscale images have clearly demonstrated that our method achieves better results than the previous state-of-the-art methods. Code will be available at `https://github.com/YeunkSuzy/Dual_Domain_Learning`.

Keywords: JPEG Artifacts Removal · Dual-Domain Learning · Fourier Transform

1 Introduction

JPEG [26], based on the Discrete Cosine Transform (DCT) [2], is one of the most widely used image compression algorithms due to its extremely high compression ratio. In JPEG, the image is divided into 8 × 8 blocks and each block is encoded separately. The DCT is then applied to each block, followed by quantization and

Guang Yang and Lu Lin are contributed equally to this work. Corresponding author: Chen Wu entropy coding. After these steps, we can obtain a good quality of the coded images with a small size. However, this comes with the loss of information, and complex artifacts inevitably appear in the compressed images. This image degradation not only causes visual discomfort, but also affects the performance of other image processing tasks, such as object detection, image super-resolution, and so on.

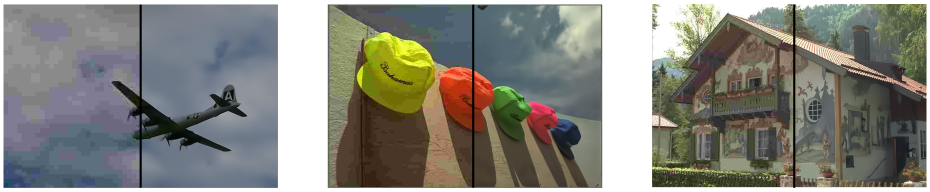

Fig. 1. Visual comparisons. The left side of the image is the JPEG compressed image and the right side is the image that has been reconstructed by our method.

In order to reduce the impact of JPEG compression artifacts, many methods have been proposed. Before deep learning was widely used in computer vision tasks, most methods solved the problem by designing a specific filter [10,23], but they were usually limited to solving specific artifacts. In recent years, with the rapid development of deep learning, JPEG artifacts removal methods based on convolutional neural networks (CNNs) [31,5,7,19,16,39,28,4,14] have prevailed and achieved better performance. However, most of the existing CNN-based methods primarily exploit the spatial information and neglect the distinguished frequency information. As we all know, JPEG compression actually occurs in the frequency domain by the DCT. Thus, exploring the effective solutions for the JPEG artifacts removal in the frequency domain is necessary.

In this paper, we explore the manifestation of JPEG compression artifacts in the frequency domain. We can transform the spatial domain image to the frequency domain by the fast Fourier transform (see Fig 2 (a)). Then, we restore the phase spectrum and amplitude spectrum separately and reconstruct the image by applying the inverse fast Fourier transform [32]. From Fig 2 (b), we can see that: (1) the Fourier phase spectrum preserves important visual structures, while the amplitude spectrum contains low-level features. (2) After JPEG compression, the Fourier phase spectrum loses some high-frequency information, while the Fourier amplitude spectrum becomes slightly blurred.

Based on the above observation, we propose a novel Dual-Domain Learning Network (D2LNet) for JPEG artifacts removal. We address JPEG artifacts removal by jointly exploring the information in the spatial and frequency domains. In order to utilize the information in both frequency domain and spatial domain effectively, we propose two core modules, namely Amplitude Correction Module (ACM) and Phase Correction Module (PCM), which are composed of multiple Amplitude Correction Blocks (ACB) and Phase Correction Blocks (PCB), re-

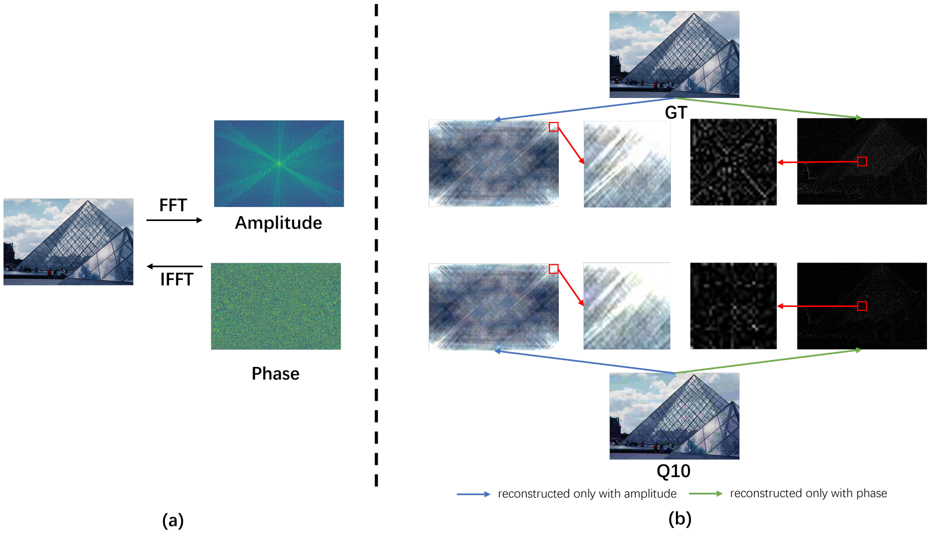
FFT
IFFT
Amplitude
Phase
GT
Q10
reconstructed only with amplitude
reconstructed only with phase
(a)
(b)

Fig. 2. Fourier transform and Fourier reconstructions. From (b), we can see that: (1) the Fourier phase spectrum preserves the important visual structures, while the amplitude spectrum contains low-level features. (2) After JPEG compression, the Fourier phase spectrum loses some high-frequency information, while the Fourier amplitude spectrum becomes slightly blurred.

spectively. Specifically, the ACM restores the amplitude spectrum of degraded images to remove JPEG artifacts, and the PCM restores the phase spectrum information to refine the high-frequency information. The qualitative and quantitative experimental results on the benchmarks show that the proposed method is effective compared to state-of-the-art methods.

2 Related Work

2.1 JPEG Artifacts Removal

JPEG compression can be represented by a formula:

$$\mathbf{Y} = \mathcal{D}(\hat{\mathbf{X}}; QF), \tag{1}$$

where $\hat{\mathbf{X}}$ and $\mathbf{Y}$ denote the original uncompressed image and the compressed image respectively, and $\mathcal{D}$ stands for the compression algorithm, and QF represents the *quality factor* determined and used for adjusting the degree of compression. Removing unwanted image artifacts which might appear on the compressed image $\mathbf{Y}$ is what we want to do. Hopefully, the restored image has a much-improved image quality, as close to $\hat{\mathbf{X}}$ as possible; that is,

$$\mathbf{X} = \mathcal{N}(\mathbf{Y}) \approx \hat{\mathbf{X}}, \tag{2}$$

where $\mathcal{N}$ means the neural network on $\mathbf{Y}$ to reconstruct or restore a very high-quality image $\mathbf{X}$ that is close to the ground-truth image $\hat{\mathbf{X}}$.

Significant progress has recently been made in reducing JPEG artifacts through the application of deep convolutional neural networks. ARCNN [7] is a relatively shallow network that first uses CNN to solve this problem. RED-Net [21] designs a deep encoding-decoding structure to exploit the rich dependencies of deep features. RNAN [37] incorporates both local and non-local attention mechanisms into its learning process, thereby enhancing its ability to represent complex relationships within images. This approach has demonstrated promising results in various image restoration tasks, such as image denoising, reducing compression artifacts, and improving image super-resolution. FBCNN [16] is a flexible blind CNN which can predict the adjustable quality factor to control the trade-off between artifacts removal and details preservation. Some GAN-based JPEG artifacts reduction works [12,13] also have good performance because they are able to produce more realistic details than MSE or SSIM [30] based networks.

2.2 Spatial-Frequency Interaction

There are several frequency domain learning methods [9,22,20,6,18,15] have achieved good results in different tasks such as image classification. For JPEG artifacts removal, learning in the frequency domain is critical because the JPEG compression is actually performed in the frequency domain using the DCT. MWCNN [19] uses wavelet to expand the receptive field to achieve image restoration. D3 [29] introduces a DCT domain prior to facilitating the JPEG artifacts removal. DWCNN [36] also removes the JPEG artifacts in the DCT domain. The DCT is a special case of the FFT, and the FFT can produce an accurate representation of the frequency domain. Although there is currently no existing work that utilizes FFT to remove JPEG artifacts, it is worth exploring the use of FFT for this purpose.

3 Method

In this section, we begin by presenting the fundamental characteristics of the Fourier transform, which hold significant relevance in comprehending our research. Subsequently, we provide a comprehensive elaboration on the proposed model and its corresponding loss function.

3.1 Fourier Transform

The Fourier transform operation, denoted by $\mathcal{F}$, allows us to convert an image $\mathbf{X}$ from the spatial domain to the frequency domain. In our work, we independently apply the Fourier transform to each channel of the image.

For a given image $\mathbf{X}$ with dimensions $H \times W$, where H represents the height and W represents the width, the Fourier transform $\mathcal{F}(\mathbf{X})(u, v)$ at frequency

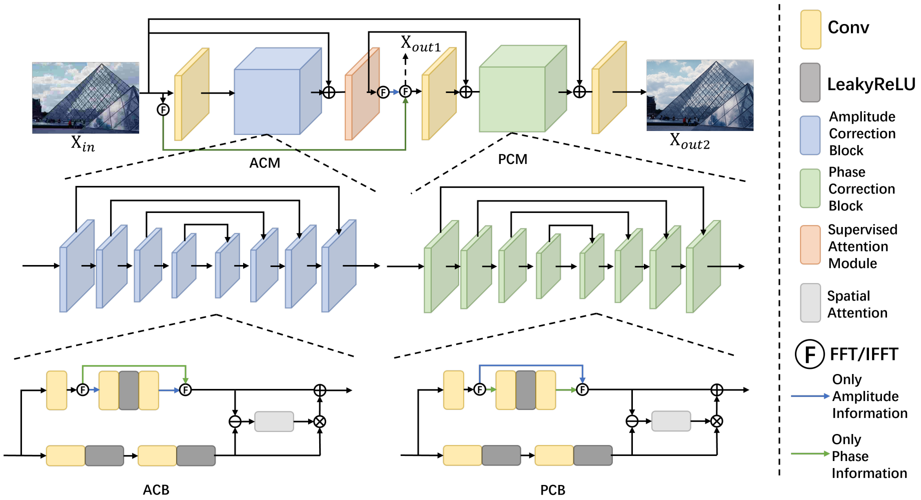

Fig. 3. The architecture of our network, which consists of two main modules: the Amplitude Correction Module (ACM) and the Phase Correction Module (PCM). Specifically, the ACM restores the amplitude spectrum of degraded images to remove JPEG artifacts, and the PCM restores the phase spectrum information to refine the high-frequency information.

domain coordinates u and v is computed according to the following equation:

$$\mathcal{F}(\mathbf{X})(u,v) = \frac{1}{\sqrt{HW}} \sum_{h=0}^{H-1} \sum_{w=0}^{W-1} \mathbf{X}(h,w) e^{-j2\pi(\frac{h}{H}u + \frac{w}{W}v)}, \tag{3}$$

It is noteworthy that the Fourier transform can be efficiently implemented using the Fast Fourier Transform (FFT) algorithm.

After performing the Fourier transform on an image $\mathbf{X}$, we can define the amplitude and frequency components as follows:

$$\mathcal{A}(\mathbf{X})(u,v) = \left[R^2(\mathbf{X})(u,v) + I^2(\mathbf{X})(u,v)\right]^{\frac{1}{2}}, \tag{4}$$

$$\mathcal{P}(\mathbf{X})(u,v) = \arctan\left[\frac{I(\mathbf{X})(u,v)}{R(\mathbf{X})(u,v)}\right]. \tag{5}$$

Here, $R(\mathbf{X})$ and $I(\mathbf{X})$ represent the real and imaginary parts of the Fourier transform of image $\mathbf{X}$. The components $\mathcal{A}(\mathbf{X})$ and $\mathcal{P}(\mathbf{X})$ correspond to the amplitude and phase of the image, respectively.

In general, the phase component of an image captures its fundamental structure and semantic information, while the amplitude component represents low-level details such as style. In the context of JPEG compression, the spatial structure of the image is compromised, and high-frequency information is significantly lost during the compression process. As a result, both the phase and amplitude suffer varying degrees of degradation.

Consequently, restoration in the frequency domain offers the advantage of separating the restoration of spatial structure and high-frequency information. Additionally, leveraging the spectral convolution theorem, image processing in the frequency domain inherently provides a global receptive field. This characteristic facilitates the capture of global information in the image.

3.2 Network Framework

Our network architecture, depicted in Figure 3, consists of two main modules: the amplitude correction module and the phase correction module, which yield two outputs, $\mathbf{X_{out1}}$ and $\mathbf{X_{out2}}$ respectively. The purpose of the amplitude correction module is to mitigate the effects of JPEG compression, while the phase correction module focuses on restoring the fundamental image details, bringing them closer to the ground truth image. To define an input image $\mathbf{X_{in}}$, we initially apply convolution to project it into the feature space. The resulting feature map is then passed through the amplitude correction module, resulting in the reconstructed feature map. Subsequently, we employ SAM [33] to derive $\mathbf{X_{out1}}$ and $\mathbf{X_{amp}}$, where $\mathbf{X_{amp}}$ is the output of the phase correction module. By replacing the amplitude of $\mathbf{X_{in}}$ with that of $\mathbf{X_{out1}}$, we obtain $\mathbf{X_{inv}}$. We concatenate $\mathbf{X_{inv}}$ and $\mathbf{X_{amp}}$, transmitting them to the phase correction module for further refining the high-frequency details of the image. Finally, we obtain the output $\mathbf{X_{out2}}$ through channel modulation. This restoration approach effectively separates the reduction of JPEG blocking artifacts from the restoration of image details.

Amplitude Correction Module. As shown in Fig 3, the amplitude correction module comprises **n** amplitude correction blocks, which aim to alleviate the influence of JPEG blocking artifacts by rectifying the amplitude. Various amplitude reconstruction units establish direct connections, known as skip links, between the lower and upper layers of the network, thereby preserving essential details. Each amplitude correction unit is composed of two branches, enabling both spatial and frequency domain learning. These branches interact in a dual domain manner, leveraging the spatial convolution's local characteristics and the frequency domain's global features. Specifically, given the feature $\mathbf{F_{Xi}}$, the amplitude correction unit can be defined as follows:

$$\mathbf{F_{spa}} = Conv(\mathbf{F_{Xi}}), \tag{6}$$

$$\mathcal{A}(F_{Xi}), \mathcal{P}(F_{Xi}) = \mathcal{F}(F_{Xi}), \tag{7}$$

$$\mathbf{F_{fre}} = \mathcal{F}^{-1}(Conv(\mathcal{A}(F_{Xi})), \mathcal{P}(F_{Xi})), \tag{8}$$

$$\mathbf{F_{dif}} = \mathbf{F_{spa}} - \mathbf{F_{fre}}, \tag{9}$$

$$\mathbf{F_{out}} = SA(F_{dif}) \cdot F_{spa} + F_{fre}. \tag{10}$$

Here, $conv(.)$ denotes a sequence of convolutional operations followed by rectified linear units (ReLU). The Fourier transform and inverse Fourier transform are represented by $\mathcal{F}(.)$ and $\mathcal{F}^{-1}(.)$, respectively. The term $SA(.)$ denotes the spatial attention mechanism [27]. In this process, the input feature $\mathbf{F_{Xi}}$ is convolved to obtain the refined feature $\mathbf{F_{spa}}$. Additionally, the amplitude and phase components, $\mathcal{A}(F_{Xi})$ and $\mathcal{P}(F_{Xi})$, are obtained by applying Fourier transform to $\mathbf{F_{Xi}}$. While keeping the phase constant, the amplitude features are reconstructed, resulting in $\mathbf{F_{fre}}$ through inverse Fourier transform. To facilitate dual domain interaction, we subtract $\mathbf{F_{fre}}$ from $\mathbf{F_{spa}}$, apply spatial attention to obtain significant weights, and finally combine the weighted fusion of $\mathbf{F_{spa}}$ and $\mathbf{F_{fre}}$ to obtain the output.

Phase Correction Module. Upon passing through the Amplitude Correction Module (ACM), $\mathbf{X_{in}}$ yields the outputs $\mathbf{X_{amp}}$ and $\mathbf{X_{out1}}$ through the Supervised Attention Module (SAM) [33]. The phase of $\mathbf{X_{amp}}$ is then substituted with the phase of $\mathbf{X_{in}}$, resulting in $\mathbf{X_{inv}}$. Subsequently, $\mathbf{X_{inv}}$ and $\mathbf{X_{amp}}$ are concatenated along the channel dimension and transmitted to the Phase Correction Module.

The Phase Correction Module comprises n phase correction blocks, aiming to enhance the high-frequency details of the image by restoring the phase component. The implementation methodology of each phase correction block is identical to that of the amplitude correction block, with the sole distinction being the preservation of the amplitude and solely addressing the phase component. Given the outputs $\mathbf{X_{amp}}$ and $\mathbf{X_{out1}}$ from the SAM, this stage can be defined by the following process:

$$\mathcal{A}(X_{out1}), \mathcal{P}(X_{out1}) = \mathcal{F}(X_{out1}), \tag{11}$$

$$\mathcal{A}(X_{in}), \mathcal{P}(X_{in}) = \mathcal{F}(X_{in}), \tag{12}$$

$$\mathbf{X_{inv}} = \mathcal{F}^{-1}(\mathcal{A}(X_{out1}), \mathcal{P}(X_{in})), \tag{13}$$

$$\mathbf{X_{out2}} = PCM(concat(\mathbf{X_{inv}}, \mathbf{X_{amp}})). \tag{14}$$

3.3 Loss Function

Our loss function consists of three components: spatial loss, amplitude loss, and phase loss, which collectively guide the reconstruction process.

Given the output X_{out2} and the ground truth image GT, the spatial loss $\mathcal{L}spa$ is defined as the L1 distance between the two:

$$\mathcal{L}spa = ||X_{out2} - GT||_1, \tag{15}$$

To guide the amplitude reconstruction module, we utilize the amplitude loss. Specifically, for X_{out1} and the ground truth image GT, the amplitude loss $\mathcal{L}amp$ is computed as:

$$\mathcal{A}(X_{out1}), \mathcal{P}(X_{out1}) = \mathcal{F}(X_{out1}), \tag{16}$$

$$\mathcal{A}(GT), \mathcal{P}(GT) = \mathcal{F}(GT), \tag{17}$$

$$\mathcal{L}amp = ||\mathcal{A}(X_{out1}) - \mathcal{A}(GT)||_1. \tag{18}$$

Similarly, the phase loss is employed to guide the reconstruction of spatial details in the image. Given X_{out2} and the ground truth image GT, the phase loss $\mathcal{L}pha$ is computed as follows:

$$\mathcal{A}(X_{out2}), \mathcal{P}(X_{out2}) = \mathcal{F}(X_{out2}), \tag{19}$$

$$\mathcal{A}(GT), \mathcal{P}(GT) = \mathcal{F}(GT), \tag{20}$$

$$\mathcal{L}pha = ||\mathcal{P}(X_{out2}) - \mathcal{P}(GT)||_1. \tag{21}$$

The total loss $\mathcal{L}$ is the weighted sum of each component:

$$\mathcal{L} = \mathcal{L}spa + \alpha * (\mathcal{L}amp + \mathcal{L}_{pha}), \tag{22}$$

Here, α is set to 0.05 based on empirical observations.

Table 1. PSNR/SSIM/PSNR-B results of our method comparaed to other nine methods in three grayscale datasets, with the best outcomes being highlighted in red.

Dataset	QF	JPEG			ARCNN			DNCNN			MWCNN			DCSC		
LIVE1	10	27.77	0.773	25.33	28.96	0.808	28.68	29.19	0.812	28.90	29.69	0.825	29.32	29.34	0.818	29.01
	20	30.07	0.851	27.57	31.29	0.873	30.76	31.59	0.880	31.07	32.04	0.889	31.51	31.70	0.883	31.18
	30	31.41	0.885	28.92	32.67	0.904	32.14	32.98	0.909	32.34	33.45	0.915	32.80	33.07	0.911	32.43
BSD500	10	27.80	0.768	25.10	29.10	0.804	28.73	29.21	0.809	28.80	29.61	0.820	29.14	29.32	0.813	28.91
	20	30.05	0.849	27.22	31.28	0.870	30.55	31.53	0.878	30.79	31.92	0.885	31.15	31.63	0.880	30.92
	30	31.37	0.884	28.53	32.67	0.902	31.94	32.90	0.907	31.97	33.30	0.912	32.34	32.99	0.908	32.08
Classic5	10	27.82	0.760	25.21	29.03	0.793	28.76	29.40	0.803	29.13	30.01	0.820	29.59	29.62	0.810	29.30
	20	30.12	0.834	27.50	31.15	0.852	30.59	31.63	0.861	31.19	32.16	0.870	31.52	31.81	0.864	31.34
	30	31.48	0.867	28.94	32.51	0.881	31.98	32.91	0.886	32.38	33.43	0.893	32.63	33.06	0.888	32.49

Dataset	QF	RNAN			RDN			QGAC			FBCNN			Ours		
LIVE1	10	29.63	0.824	29.13	29.70	0.825	29.37	29.51	0.825	29.13	29.75	0.827	29.40	30.08	0.840	30.06
	20	32.03	0.888	31.12	32.10	0.889	31.29	31.83	0.888	31.25	32.13	0.889	31.57	32.47	0.898	32.42
	30	33.45	0.915	32.22	33.54	0.916	32.62	33.20	0.914	32.47	33.54	0.916	32.83	33.91	0.924	33.85
BSD500	10	29.08	0.805	28.48	29.24	0.808	28.71	29.46	0.821	28.97	29.67	0.821	29.22	29.61	0.810	29.53
	20	31.25	0.875	30.27	31.48	0.879	30.45	31.73	0.884	30.93	32.00	0.885	31.19	32.04	0.890	31.67
	30	32.70	0.907	31.33	32.83	0.908	31.60	33.07	0.912	32.04	33.37	0.913	32.39	33.41	0.916	32.97
Classic5	10	29.96	0.819	29.42	30.03	0.819	29.59	29.84	0.812	29.43	30.12	0.822	29.80	31.18	0.838	31.11
	20	32.11	0.869	31.26	32.19	0.870	31.53	31.98	0.869	31.37	32.31	0.872	31.74	33.41	0.886	33.27
	30	33.38	0.892	32.35	33.46	0.893	32.59	33.22	0.892	32.42	33.54	0.894	32.78	34.71	0.907	34.49

4 Experiments

4.1 Experimental Datasets and Implementation Details

In our experiments, we employ DIV2K [1] and Flickr2K [25] as our training data. During training, we randomly crop 256 × 256 patches from the images. In addition, we have compressed them with JPEG with different quality factors Q = 10, 20 and 30. To optimize the parameters of D2LNet, we adopt the Adam optimizer [17] with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. We train our model on one NVIDIA GeForce GTX 3060 GPU by using PyTorch.

During testing, we evaluate the performance of our model on Classic5 [34], LIVE1 [24], and the test set of BSDS500 [3] for grayscale images. For color images, we do not use the Classic5 but the ICB [8] instead. We use PSNR, SSIM (structural similarity) [30], and PSNR-B (specially designed for JPEG artifacts removal) to quantitatively assess the performance of our JPEG artifacts removal model.

4.2 Results

To evaluate the effectiveness of our model, we conducted experiments on both grayscale and color images. We use the Y channel of YCbCr space for grayscale image comparison, and the RGB channels for color image comparison.

Grayscale JPEG Image Restoration. We first evaluate the effect of our model on the Y-channel JPEG compressed images. For benchmarking purposes, we chose ARCNN [7], DNCNN [35], MWCNN [19], DCSC [11], RNAN [37], RDN [38], QGAC [8], and the powerful FBCNN [16] as reference methods. Table 1 presents the comparison results, with the superior outcomes highlighted in red. Our method consistently achieved the best performance across multiple datasets, as evaluated using three assessment metrics. This observation underscores the considerable potential of incorporating frequency domain information in JPEG restoration.

Color JPEG Image Restoration. To further showcase the efficacy of our model, we conducted restoration experiments on color datasets. Considering the increased complexity of color image restoration, we selected QGAC [8] and FBCNN [16] methods for comparison. As shown in Table 4.2, the results clearly demonstrate the superiority of our approach in color image restoration, reinforcing the robustness of our model and the significance of frequency domain information.

Table 2. PSNR/SSIM/PSNR-B results of different methods on the three color datasets, with the best outcomes being highlighted in red.

Dataset	QF	JPEG			QGAC			FBCNN			Ours		
LIVE1	10	25.69	0.743	24.20	27.62	0.804	27.43	27.77	0.803	27.51	27.82	0.805	27.80
	20	28.06	0.826	26.49	29.88	0.868	29.56	30.11	0.868	29.70	30.14	0.871	30.11
	30	29.37	0.861	27.84	31.17	0.896	30.77	31.43	0.897	30.92	31.49	0.899	31.33
BSD500	10	25.84	0.741	24.13	27.74	0.802	27.47	27.85	0.799	27.52	27.66	0.778	27.61
	20	28.21	0.827	26.37	30.01	0.869	29.53	30.14	0.867	29.56	30.15	0.869	29.75
	30	29.57	0.865	27.72	31.33	0.898	30.70	31.45	0.897	30.72	31.52	0.897	30.97
ICB	10	29.44	0.757	28.53	32.06	0.816	32.04	32.18	0.815	32.15	33.02	0.829	32.97
	20	32.01	0.806	31.11	34.13	0.843	34.10	34.38	0.844	34.34	34.57	0.847	34.49
	30	33.95	0.831	32.35	35.07	0.857	35.02	35.41	0.857	35.35	35.55	0.856	35.44

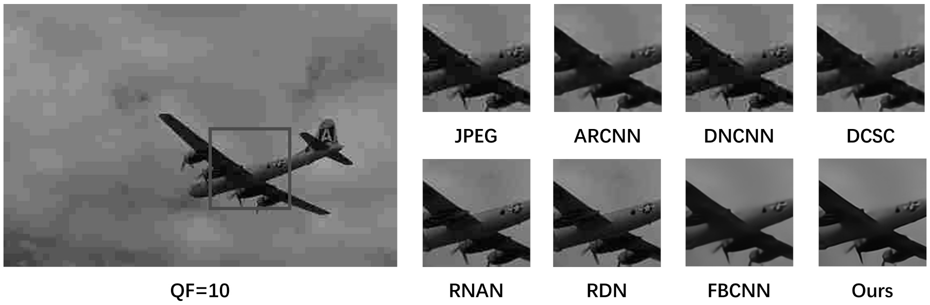

Fig. 4. Visual comparisons of JPEG image "BSD: 3096" with $QF = 10$.

4.3 Ablation Studies

To provide additional insights into the functionality of our proposed module, we performed ablation experiments on three color image datasets with a quantization factor of Q10. The ablation experiments were designed to evaluate the impact of the Amplitude Correction Block and the Phase Correction Block. Two groups of experiments were conducted, wherein we replaced these units with resblocks having similar parameter settings. This allowed us to assess the effects of eliminating the amplitude reconstruction and phase reconstruction functionalities.

Amplitude Correction Block. The Amplitude Correction Block primarily focuses on the image's amplitude, aiming to mitigate the effects of JPEG compression and alleviate block artifacts. In the first set of experiments, we substituted the amplitude reconstruction unit with a resblock, as indicated in the first row of the Table 3. It is evident from the results that various metrics exhibit a certain degree of decline, providing evidence that processing the amplitude effectively restores the compression-induced degradation in the image.

Phase Correction Block. The Phase Correction Block is designed to restore fine details in the image, bringing the texture edges closer to the ground truth. In the second set of experiments, we substituted the phase reconstruction unit with a resblock, as presented in the second row of the Table 3. Upon removing the phase reconstruction unit, the evaluation metrics exhibited a decrease across all three datasets. This observation underscores the significance of phase information in the image restoration task.

Table 3. The results of the ablation experiments conducted on the three datasets.

CONFIG	ACB	PCB	BSD500			LIVE1			ICB		
			PSNR	SSIM	PSNR-B	PSNR	SSIM	PSNR-B	PSNR	SSIM	PSNR-B
(I)	×	✓	27.54	0.776	27.49	27.72	0.801	27.69	31.74	0.813	31.73
(II)	✓	×	27.47	0.773	27.44	27.71	0.799	27.63	31.75	0.813	31.74
Ours	✓	✓	27.66	0.778	27.61	27.82	0.805	27.80	33.02	0.829	32.97

5 Conclusions

In this paper, we propose a Dual-Domain Learning Network for JPEG artifacts removal (D2LNet). In contrast to previous JPEG artifacts removal methods performed directly in the spatial domain, we combined the information from the frequency domain. The Amplitude Correction Module (ACM) and Phase Correction Module (PCM) have effectively achieved information interaction between the spatial and frequency domains. Extensive experiments on the grayscale JPEG images and the color JPEG images demonstrate the effectiveness and generalizability of our proposed D2LNet. Nevertheless, our approach is sensitive to the quality factor (QF), which is clearly a drawback. We should delve into the research on utilizing frequency domain information to achieve flexible blind JPEG artifacts removal.

Acknowledgements This work is supported by National MCF Energy R&D Program of China (Grant No: 2018YFE0302100).

References

1. Agustsson, E., Timofte, R.: Ntire 2017 challenge on single image super-resolution: Dataset and study. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops. pp. 126–135 (2017)
2. Ahmed, N., Natarajan, T., Rao, K.R.: Discrete cosine transform. IEEE transactions on Computers **100**(1), 90–93 (1974)
3. Arbelaez, P., Maire, M., Fowlkes, C., Malik, J.: Contour detection and hierarchical image segmentation. IEEE transactions on pattern analysis and machine intelligence **33**(5), 898–916 (2010)
4. Cavigelli, L., Hager, P., Benini, L.: Cas-cnn: A deep convolutional neural network for image compression artifact suppression. In: 2017 International Joint Conference on Neural Networks (IJCNN). pp. 752–759. IEEE (2017)
5. Chen, Y., Pock, T.: 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 (2016)
6. Chi, L., Jiang, B., Mu, Y.: Fast fourier convolution. Advances in Neural Information Processing Systems **33**, 4479–4488 (2020)
7. Dong, C., Deng, Y., Loy, C.C., Tang, X.: Compression artifacts reduction by a deep convolutional network. In: Proceedings of the IEEE international conference on computer vision. pp. 576–584 (2015)
8. Ehrlich, M., Davis, L., Lim, S.N., Shrivastava, A.: Quantization guided jpeg artifact correction. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VIII 16. pp. 293–309. Springer (2020)
9. Ehrlich, M., Davis, L.S.: Deep residual learning in the jpeg transform domain. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 3484–3493 (2019)
10. Foi, A., Katkovnik, V., Egiazarian, K.: Pointwise shape-adaptive dct for high-quality denoising and deblocking of grayscale and color images. IEEE transactions on image processing **16**(5), 1395–1411 (2007)

11. Fu, X., Zha, Z.J., Wu, F., Ding, X., Paisley, J.: Jpeg artifacts reduction via deep convolutional sparse coding. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2501–2510 (2019)
12. Galteri, L., Seidenari, L., Bertini, M., Del Bimbo, A.: Deep generative adversarial compression artifact removal. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 4826–4835 (2017)
13. Galteri, L., Seidenari, L., Bertini, M., Del Bimbo, A.: Deep universal generative adversarial compression artifact removal. IEEE Transactions on Multimedia **21**(8), 2131–2145 (2019)
14. Guo, J., Chao, H.: Building dual-domain representations for compression artifacts reduction. In: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14. pp. 628–644. Springer (2016)
15. Huang, J., Liu, Y., Zhao, F., Yan, K., Zhang, J., Huang, Y., Zhou, M., Xiong, Z.: Deep fourier-based exposure correction network with spatial-frequency interaction. In: European Conference on Computer Vision. pp. 163–180. Springer (2022)
16. Jiang, J., Zhang, K., Timofte, R.: Towards flexible blind jpeg artifacts removal. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4997–5006 (2021)
17. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
18. Li, Z., Kovachki, N., Azizzadenesheli, K., Liu, B., Bhattacharya, K., Stuart, A., Anandkumar, A.: Fourier neural operator for parametric partial differential equations. arXiv preprint arXiv:2010.08895 (2020)
19. Liu, P., Zhang, H., Zhang, K., Lin, L., Zuo, W.: Multi-level wavelet-cnn for image restoration. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops. pp. 773–782 (2018)
20. Liu, T., Cheng, J., Tan, S.: Spectral bayesian uncertainty for image super-resolution. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 18166–18175 (2023)
21. Mao, X., Shen, C., Yang, Y.B.: Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections. Advances in neural information processing systems **29** (2016)
22. Mao, X., Liu, Y., Shen, W., Li, Q., Wang, Y.: Deep residual fourier transformation for single image deblurring. arXiv preprint arXiv:2111.11745 (2021)
23. Ren, J., Liu, J., Li, M., Bai, W., Guo, Z.: Image blocking artifacts reduction via patch clustering and low-rank minimization. In: 2013 Data Compression Conference. pp. 516–516. IEEE (2013)
24. Sheikh, H.: Live image quality assessment database release 2. http://live. ece. utexas. edu/research/quality (2005)
25. Timofte, R., Agustsson, E., Van Gool, L., Yang, M.H., Zhang, L.: Ntire 2017 challenge on single image super-resolution: Methods and results. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops. pp. 114–125 (2017)
26. Wallace, G.K.: The jpeg still picture compression standard. IEEE transactions on consumer electronics **38**(1), xviii–xxxiv (1992)
27. Wang, H., Fan, Y., Wang, Z., Jiao, L., Schiele, B.: Parameter-free spatial attention network for person re-identification. arXiv preprint arXiv:1811.12150 (2018)
28. Wang, X., Fu, X., Zhu, Y., Zha, Z.J.: Jpeg artifacts removal via contrastive representation learning. In: European Conference on Computer Vision. pp. 615–631. Springer (2022)

29. Wang, Z., Liu, D., Chang, S., Ling, Q., Yang, Y., Huang, T.S.: D3: Deep dual-domain based fast restoration of jpeg-compressed images. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2764–2772 (2016)
30. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing **13**(4), 600–612 (2004)
31. Xu, L., Ren, J.S., Liu, C., Jia, J.: Deep convolutional neural network for image deconvolution. Advances in neural information processing systems **27** (2014)
32. Xu, Q., Zhang, R., Zhang, Y., Wang, Y., Tian, Q.: A fourier-based framework for domain generalization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14383–14392 (2021)
33. Zamir, S.W., Arora, A., Khan, S., Hayat, M., Khan, F.S., Yang, M.H., Shao, L.: Multi-stage progressive image restoration. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 14821–14831 (2021)
34. Zeyde, R., Elad, M., Protter, M.: On single image scale-up using sparse-representations. In: Curves and Surfaces: 7th International Conference, Avignon, France, June 24-30, 2010, Revised Selected Papers 7. pp. 711–730. Springer (2012)
35. Zhang, K., Zuo, W., Chen, Y., Meng, D., Zhang, L.: Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. IEEE transactions on image processing **26**(7), 3142–3155 (2017)
36. Zhang, X., Yang, W., Hu, Y., Liu, J.: Dmcnn: Dual-domain multi-scale convolutional neural network for compression artifacts removal. In: 2018 25th IEEE international conference on image processing (icip). pp. 390–394. IEEE (2018)
37. Zhang, Y., Li, K., Li, K., Zhong, B., Fu, Y.: Residual non-local attention networks for image restoration. arXiv preprint arXiv:1903.10082 (2019)
38. Zhang, Y., Tian, Y., Kong, Y., Zhong, B., Fu, Y.: Residual dense network for image restoration. IEEE Transactions on Pattern Analysis and Machine Intelligence **43**(7), 2480–2495 (2020)
39. Zini, S., Bianco, S., Schettini, R.: Deep residual autoencoder for blind universal jpeg restoration. IEEE Access **8**, 63283–63294 (2020)