DAQE: Enhancing the Quality of Compressed Images by Finding the Secret of Defocus

Qunliang Xing, Graduate Student Member, IEEE, Mai Xu, Senior Member, IEEE, Xin Deng, Member, IEEE, and Yichen Guo

Abstract—Image defocus is inherent in the physics of image formation caused by the optical aberration of lenses, providing plentiful information on image quality. Unfortunately, the existing quality enhancement approaches for compressed images neglect the inherent characteristic of defocus, resulting in inferior performance. This paper finds that in compressed images, the significantly defocused regions are with better compression quality and two regions with different defocus values possess diverse texture patterns. These findings motivate our defocus-aware quality enhancement (DAQE) approach. Specifically, we propose a novel dynamic region-based deep learning architecture of the DAQE approach, which considers the region-wise defocus difference of compressed images in two aspects. (1) The DAQE approach employs fewer computational resources to enhance the quality of significantly defocused regions, while more resources on enhancing the quality of other regions; (2) The DAQE approach learns to separately enhance diverse texture patterns for the regions with different defocus values, such that texture-wise one-on-one enhancement can be achieved. Extensive experiments validate the superiority of our DAQE approach in terms of quality enhancement and resource-saving, compared with other state-of-the-art approaches.

Index Terms—Image defocus, quality enhancement, compressed image, deep learning.

1 INTRODUCTION

Nowadays, we are embracing an era of the explosive growth of images. According to Domo statistics [1], Facebook stored and transmitted around 147,000 images per minute in 2020; similar situations also apply to other Internet servers, such as WeChat and Twitter. To store and transmit such a huge amount of images, several lossy image compression standards, e.g., joint photographic experts group (JPEG) [2], JPEG 2000 [3], and high-efficiency video coding with main still image profile (HEVC-MSP)/better portable graphics (BPG) [4], [5], have been successfully developed to save transmission bandwidth and storage cost. However, the compressed images inevitably suffer from compression artifacts, e.g., ringing, blocking, and blurring effects [6], thus degrading the quality of user experience (QoE) of images [7], [8].

This paper proposes enhancing the quality of compressed images by taking into account the characteristic of image defocus, which is a blurring effect caused by the optical aberrations of lenses. Specifically, only regions close to the focal plane, i.e., within the depth of field (DoF) [9], appear to be focused, while other regions that are far from the focal plane are blurred [10]. Given the characteristic of image defocus, there exist two main drawbacks in the state-of-the-art approaches [11], [12], [13], [14], [15], [16], [17], [18], [19], [20] for quality enhancement of compressed images.

• Q. Xing is with the School of Electronic Information Engineering and the Shen Yuan Honors College, Beihang University, Beijing, China. E-mail: xingqi@buaa.edu.cn.
• M. Xu and Y. Guo are with the School of Electronic Information Engineering, Beihang University, Beijing, China. E-mail: {maixu,gye970930}@buaa.edu.cn.
• X. Deng is with the School of Cyber Science and Technology, Beihang University, Beijing, China. E-mail: cindydeng@buaa.edu.cn.
• Corresponding author: Mai Xu.

(1) Region-wise quality agnostic. The existing approaches neglect the difference in the quality of different regions of an input image; correspondingly, they process the whole image in the same manner. However, there exists a significant region-wise quality difference in a single compressed image, in particular referring to the regions with different defocus values. (2) Region-wise texture agnostic. The existing approaches do not consider the texture difference in a compressed image. Consequently, they are not effective in enhancing diverse texture patterns, of which the diversity can also be reflected in their defocus values. Ideally, one-on-one quality enhancement should be conducted for diverse texture patterns, especially those of regions with different defocus values.

We address the above two drawbacks of existing approaches by utilizing the inherent and off-the-shelf image defocus. We obtain two findings by analyzing the defocus and quality of compressed images over the diverse 2K resolution image (DIV2K) dataset [22], as shown in Figure 1. (1) The compression quality of compressed images is highly correlated with image defocus. Specifically, in a compressed image, the significantly defocused regions have better compression quality, compared with the slightly defocused regions. Thus, the regions with different defocus values in a compressed image should be separately enhanced. (2) The regions with different defocus values tend to have diverse texture patterns. Therefore, texture-wise one-on-one enhancement can be achieved by separately enhancing regions with different defocus values.

Based on our findings, we propose a defocus-aware quality enhancement approach, named DAQE, for enhancing the quality of compressed images. The DAQE approach is equipped with a novel dynamic deep-learning-based architecture. First, the DAQE approach estimates the de-
focus map for the input image. Then, the DAQE approach conducts the patch-wise dynamic enhancement for patches with different defocus values separately. Considering that the significantly defocused patches are superior in compression quality compared with the slightly defocused ones, the DAQE approach employs fewer computational resources to enhance the quality of the significantly defocused patches and more resources on other patches to improve efficiency. Note that all patches are enhanced with a single dynamic architecture in an “easy-to-hard” manner. Also, the DAQE approach extracts diverse texture patterns for patches with different defocus values, by embedding a unique attention-based texture learner in each enhancement path. The texture learner is designed to extract texture patterns diverse in shape and intensity. This way, we can achieve texture-wise one-on-one quality enhancement with improved efficacy. Finally, we conduct extensive experiments to validate the effectiveness of our DAQE approach in terms of quality enhancement and resource saving, significantly better than other state-of-the-art approaches.

2 RELATED WORKS

2.1 Quality Enhancement of Compressed Images

During the past decade, many deep-learning-based approaches [11, 12, 13, 15, 16, 20] have been proposed for enhancing the quality of compressed images, owing to the successful development of convolutional neural networks (CNNs) [23]. Specifically, Dong et al. [11] proposed a shallow four-layer artifacts reduction CNN (AR-CNN), as the pioneer of CNNs-based quality enhancement approaches for JPEG-compressed images. Later, approaches with deeper CNN structures and the quantization prior of JPEG compression, i.e., deep dual-domain (D3) [13] and deep dual-domain CNN (DDCN) [12], were proposed to remove JPEG compression artifacts. Wang et al. [16] proposed a 10-layer deep CNN-based auto decoder (DCAD), which is the first CNNs-based quality enhancement approach for BPG-compressed images. DCAD does not utilize any coding information from codecs but surpasses most previous approaches in terms of the quality of enhanced images thanks to the effective learning structure of a much deeper network. To make a step forward, the denoising convolutional neural networks (DnCNN) [15] was proposed, which combines a 20-layer deep network with some advanced techniques of the day including the residual learning [24] and batch normalization [25]. This way, DnCNN significantly outperforms most traditional model-based approaches such as block-matching and 3-D filtering (BM3D) [26], as well as the above learning-based approaches. Most recently, Xing et al. [20] proposed a resource-efficient blind quality enhancement (RBQE) approach for both JPEG-compressed and BPG-compressed images. The RBQE approach was designed with a dynamic inference structure, such that blind yet effective quality enhancement can be achieved for compressed images. In this paper, we propose utilizing the characteristic of image defocus to facilitate the quality enhancement of compressed images.

2.2 Defocus-Aware Vision Tasks

In this section, we review defocus-aware works of related vision tasks. The characteristic of image defocus provides plentiful information about image quality, depth, objectness, saliency, etc. Hence, image defocus has been widely used in many vision tasks, e.g., image depth estimation [27, 28, 29, 30], image defocus deblurring [31, 32, 33], image saliency detection [34, 35], and image segmentation [36]. For image depth estimation, Pentland et al. [27] showed that two images formed with different apertures indicate depth information; thus, the image depth can be generated from image defocus. For image defocus deblurring, the works of [31, 32, 33] were proposed, in which the defocus kernel is estimated and then used to deblur images. For image saliency prediction, Jiang et al. [35] found that salient image regions are often photographed in focus; therefore, the estimation of image defocus maps can boost the performance of higher-level saliency prediction.
To the best of our knowledge, there exist no defocus-aware quality enhancement works for compressed images. Besides, it is unclear about the correlation between the region quality and region defocus of compressed images. In this paper, we thoroughly investigate this correlation and demonstrate that the characteristic of image defocus can significantly benefit the quality enhancement task by our proposed DAQE approach.

3 FINDINGS

This section presents our findings on how the characteristic of defocus is related to the region-wise quality and texture patterns of the compressed images. Our findings are obtained by analyzing a widely-used DIV2K dataset [22], which includes 900 images with 2K resolution. These images cover a large diversity of contents, including people (13.67%), flora and fauna (31.56%), man-made objects (19.11%), cityscapes (20.78%), and landscapes (14.89%), as shown in Figure 2. Besides, these images can also fall into the scenes of indoor (11.89%), outdoor (83.89%), and underwater (4.22%). First, to evaluate the defocus level for each image, we adopt the state-of-the-art DMENet [21] to generate a defocus map for each image. Then, to obtain compressed images, we compress all images with two compression codecs (i.e., the BPG [5] and JPEG [2] codecs) and eight settings (i.e., with a quantization parameter (QP) = 27/32/37/42 or a quality factor (QF) as 20/30/40/50). Next, to evaluate the region-wise defocus, quality, and texture patterns, we crop all images and defocus maps into non-overlapping patches with a size of 128 × 128. Finally, we calculate the average defocus value for each patch, as the patch-wise defocus value.

Finding 1: There exists a dramatic variation of the patch-wise defocus values within a single image.

Analysis: We measure the variation of the patch-wise defocus values in a single image in terms of the standard deviation (STD), coefficient of variation (CV) [27], and range. Specifically, the CV value is the ratio of the STD value to the mean value; the range value is obtained by subtracting the lowest patch-wise defocus value from the highest one within an image. As shown in Table 1, the CV value is no less than 40% for all contents, indicating a strong variation of the patch-wise defocus values. Besides, the defocus range is up to 135.84 (i.e., for the content Flora & Fauna), which is almost two times the corresponding mean value (i.e., 69.99). Similar results can be found for other contents in Table 1, implying a huge interval of the patch-wise defocus values within each image. Thus, the variation of the patch-wise defocus values within a single image is dramatic. This completes the analysis of Finding 1.

Finding 2: For a compressed image, the patches with higher defocus values tend to have better compression quality.

Analysis: We adopt two widely-used quality assessment

TABLE 1

Variation of the patch-wise defocus values within a single image.

| Content     | People | Flora & Fauna | Man-made objects | Cityscapes |
|-------------|--------|---------------|------------------|------------|
| STD         | 34.26  | 39.13         | 26.06            | 37.79      |
| Mean (%)    | 64.64  | 69.99         | 50.00            | 55.17      |
| CV (%)      | 50.16  | 54.74         | 46.67            | 61.10      |
| Range       | 129.16 | 135.84        | 107.26           | 130.67     |

Fig. 3. Correlation between patch quality and features. The “lum” and “cont” are the abbreviations of “luminance” and “contrast”.
metrics, i.e., the peak signal-to-noise ratio (PSNR) and structural similarity index measure (SSIM) [38], for measuring the compression quality. Then, for each compression setting, the Pearson correlation coefficient (PCC) [39] and Spearman’s rank correlation coefficient (SRCC) [40] values are calculated between the defocus and quality values for all patches, to validate their correlation. Results are then averaged by eight compression settings. In addition to defocus, we adopt the features of luminance, contrast, and total variation (TV) as the baseline. As shown in Figure 2, both the PSNR and SSIM values of patches are highly correlated with their corresponding defocus values. Specifically, both the PCC and SRCC values between quality and defocus are above 0.70, significantly higher than those between quality and defocus values. The texture difference is measured by the TDIM value (×10³).

4 Proposed approach
In this section, we focus on our proposed DAQE approach for enhancing the quality of compressed images. The DAQE approach aims to enhance the quality of the regions with different defocus values. Considering that these regions differ significantly in compression quality and texture patterns (as illustrated in Findings 2 and 3), we implement the DAQE approach by proposing an enhancement framework with mainly three steps as shown in Figure 6 (a), i.e., defocus estimation, attention generation, and dynamic quality enhancement.

Specifically, (1) the DAQE approach first estimates the defocus value for each image patch with a proposed defocus estimation network (DENet). (2) Then, the DAQE approach divides patches into N clusters according to their defocus values, and conducts cluster-specific texture extraction and quality enhancement. To extract the texture pattern for each patch, the DAQE approach processes the input patch with a proposed attention generation network (AGNet). The AGNet consists of a convolution head and a transformer head to extract the texture pattern with local and global attention, respectively. On the convolution head, local attention maps are generated to normalize the encoded feature of the input patch. On the transformer head, the input patch is encoded and normalized by global attention to some reference patch. On the transformer head, the input patch is encoded and normalized by global attention to some reference patch. On the transformer head, the input patch is encoded and normalized by global attention to some reference patch. On the transformer head, the input patch is encoded and normalized by global attention to some reference patch. On the transformer head, the input patch is encoded and normalized by global attention to some reference patch.

As a result, defocus can serve as a good indicator for region-wise compression quality (measured by PSNR and SSIM). More importantly, the correlation between defocus and quality is positive, implying superior compression quality for patches with higher defocus values. This completes the analysis of Finding 2.

Finding 3: For a compressed image, the texture patterns of the patches with dissimilar defocus values are more diverse, compared with those with similar defocus values.

Analysis: We cluster all patches into three clusters by the K-means clustering algorithm [41], [42] according to their defocus values. Figure 4 shows that the patches at different clusters can differ significantly in quality, which is in accord with Finding 2. Here, we further measure the average texture difference between patches at the same/different clusters. Specifically, the texture difference of two patches is measured by the Frobenius norm of the difference between their Gram matrices of Y components, named texture difference index measure (TDIM), which has been widely used in many texture-related works [43], [44]. Note that larger TDIM values indicate more diversity of the texture patterns between two patches. As shown in Figure 5, the TDIM values between patches at two different clusters are much larger than those between patches at the same cluster. For example, for images compressed at QP = 37, the average TDIM value between two patches at the first cluster is \(2.19 \times 10^3\), significantly lower than that between two patches at clusters 1 and 3 (i.e., \(4.03 \times 10^3\)). Therefore, for a compressed image, the texture patterns of patches with similar defocus values are more diverse, compared with those with similar defocus values. This completes the analysis of Finding 3.

DQAEmODA: ENHANCING THE QUALITY OF COMPRESSED IMAGES BY FINDING THE SECRET OF DEFOCUS
4
efficiently, by taking the advantage of the inherent image defocus information.

### 4.1 Defocus Estimation

In our DAQE approach, we design the DENet to estimate a defocus map \( \mathbf{M} \) for the input compressed image \( \mathbf{I}_n \). As shown in Figure 3(b), the DENet first adopts a series of residual blocks \( R_{\text{clean}} \) to remove the severe compression artifacts of \( \mathbf{I}_n \). As a result, the pre-cleaned feature \( \mathbf{F}_{\text{clean}} \) is generated from \( \mathbf{I}_n \) and is then encoded to be \( \mathbf{F}_{\text{VGG}} \) by a VGG [46] encoder, denoted by \( E_{\text{VGG}} \). Here, \( E_{\text{VGG}} \) is pre-trained on ImageNet [47] because the pre-training on a large-scale dataset can facilitate the cross-domain learning of image defocus estimation (i.e., from the image domain to the defocus feature domain), as inspired by DMENet [21]. Finally, a U-Net [48] based decoder, denoted by \( D_{\text{U-Net}} \), is adopted followed by a series of residual blocks \( R_{\text{out}} \), for generating the defocus map \( \mathbf{M} \) from \( \mathbf{F}_{\text{VGG}} \). Mathematically, we can obtain the defocus map \( \mathbf{M} \) for the input compressed image \( \mathbf{I}_n \) as follows,

\[
\mathbf{M} = R_{\text{out}} (D_{\text{U-Net}} (E_{\text{VGG}} (R_{\text{clean}} (\mathbf{I}_n)))) ,
\]

where \( \mathbf{M} \) and \( \mathbf{I}_n \) are with the same resolution.

Recall that the patch-wise defocus value is the average defocus value for each patch. Therefore, we can finally obtain the patch-wise defocus value for each patch, by first dividing \( \mathbf{M} \) into non-overlapping \( S \times S \) patches together with \( \mathbf{I}_n \), and then taking the average of the corresponding patch of \( \mathbf{M} \).

### 4.2 Defocus-Aware Attention Generation

In our DAQE approach, we design the AGNet to extract the texture pattern for an input patch. The attention mechanism [49] has been widely used to extract texture patterns for image quality enhancement and other restoration tasks [50], [51], [52], [53], [54]. However, the various implementations of the attention mechanism in these works are conducted over the whole image, neglecting the texture diversity of different regions. As shown in Figure 7(a), the AGNet mitigates this drawback by implementing the attention mechanism for regions with different defocus values separately. It is because those regions differ significantly in texture patterns, as discussed in Finding 3. Specifically, the AGNet first divides all patches into \( N_{\text{clus}} \) clusters according to their defocus values. Then, for each input patch, the AGNet captures (1) the local attention to the input patch itself and (2) the global attention to some reference patches at the same cluster. Finally, the AGNet encodes and normalizes the input patch with both the captured local and global attention, such that the texture pattern of the input patch can be obtained.

**Patch clustering.** The AGNet first divides all patches of an input image into \( N_{\text{clus}} \) clusters according to their defocus values. First, the defocus value centers of clusters are determined by the K-means algorithm [41], [42] over a large-scale dataset. Then, every patch belongs to its closest cluster, in terms of the sum-of-squares distances between its defocus value and the center defocus values. This way, the AGNet can cluster the patches of the input image into different clusters with different defocus centers. It is worth pointing out that according to Finding 3, the patches at different clusters possess diverse texture patterns. This observation is utilized in the following attention generation.

**Local attention generation.** As aforementioned, the attention mechanism has been widely used to extract texture patterns for image quality enhancement and other restoration tasks. As inspired by the spatial adaptive normalization layer [52], our AGNet includes a texture modeling subnet (TM subnet) for each cluster of patches, to extract the texture pattern for an input patch, as illustrated in Figure 7(a). We
take the $m$-th TM subnet as an example, which processes the input patches at the $m$-th cluster, denoted by $P_m$. First, $P_m$ is convolved by residual blocks $R_{in,m}$ to obtain the encoded feature $F_{in,m}$. Then, $P_m$ is convolved by residual blocks $R_{γ,m}$ and $R_{β,m}$, such that the attention maps $F_{γ,m}$ and $F_{β,m}$ can be produced, respectively. Then, $F_{in,m}$ is element-wise multiplied by $F_{γ,m}$ and then added by $F_{β,m}$, for generating the final output $F_{LA,m}$. Mathematically, the above processes can be written as follows,

$$F_{LA,m} = R_{in,m}(P_m) \odot R_{γ,m}(P_m) + R_{β,m}(P_m).$$

Note that the patches at the same cluster share a TM subnet, i.e., with shared parameters, while those at different clusters are fed into different TM subnets, i.e., with different sets of parameters that are learned separately. It is because the texture patterns of the patches at different clusters are more diverse, compared with those at the same cluster.

**Global attention generation.** Finding 3 reveals that the texture patterns of the patches at the same cluster are more similar, compared with those of the patches at different clusters. In light of this observation, the AGNet also includes a global branch in addition to the local one for each TM subnet, to take advantage of the patches at the same cluster. As depicted in Figure 7(a), the global branch works through the following steps.

1) $N_{ref}$ patches are proposed as the global reference patches [55], which generate the key/value pairs for the queries of all patches at their same cluster. The AGNet first uniformly samples $I_m$ to generate the initial reference patches, with the spatial sampling interval of $4S$ in both directions. Assume the height and width of $I_m$ are $H$ and $W$, respectively. Then, we have $N_{ref}$ global reference patches, and $N_{ref}$ equals to $\lceil H/(4S) \rceil \times \lceil W/(4S) \rceil$.

2) The AGNet adjusts the positions $\{(x_i, y_i)\}_{i=1}^{N_{ref}}$ of the initial reference patches by adding the position offsets $\{(\Delta x_i, \Delta y_i)\}_{i=1}^{N_{ref}}$. Here, $\{(\Delta x_i, \Delta y_i)\}_{i=1}^{N_{ref}}$ can be learned in the form of an offset map $F_{offset}$ through an offset learning subnet as follows,

$$F_{offset} = C_{1 \times 1}(L_{GELU}(C_{DW}(I_m))),$$

where $C_{1 \times 1}$, $L_{GELU}$, and $C_{DW}$ denote a convolution layer with a kernel size of $1 \times 1$, a GELU activation layer [56], and a depth-wise convolution layer, respectively. Note that $F_{offset}$ is a map with a size of $\lceil H/(4S) \rceil \times \lceil W/(4S) \rceil$, in which each element denotes the offset pair $(\Delta x_i, \Delta y_i)$ for the $i$-th reference patch.

3) Given the reference patches, the AGNet calculates the query of every input patch and the key/value pairs of only the reference patches for each cluster as follows,

$$Q_m = \tilde{P}_m W_Q^m,$$

$$K_m = \tilde{P}_m W_K^m,$$

$$V_m = \tilde{P}_m W_V^m.$$

In the above equations, $\tilde{P}_m$ and $\tilde{P}_m^{(i)}$ are the flattened $P_m$ and the flattened $i$-th reference patch of $P_m$ at the $m$-th cluster, respectively; $W_Q^m$, $W_K^m$, and $W_V^m$ are the projection matrices for the query, key, and value, respectively; $N_{ref}^m$ is the number of reference patches at the $m$-th cluster; $Q_m$, $K_m^{(i)}$ and $V_m^{(i)}$ are the query of $P_m$, key of $P_m^{(i)}$, and value of $P_m^{(i)}$, respectively. If $N_{ref}^m$ equals 0, we choose the reference patch at the neighboring cluster with the defocus value closest to the center of this cluster.
4) The AGNet performs the multi-head attention between \( Q_m \) and each \( (K^{(i)}_m, V^{(i)}_m) \) pair. The attention output \( Z_m \) of each attention head can be formulated as,

\[
Z_m = \sum_{i=1}^{N_m} \sigma \left( Q_m K^{(i)\top}_m / \sqrt{d} + B \right) V^{(i)}_m. \tag{7}
\]

In the above equation, \( \sigma \) denotes the softmax function; \( d \) is the dimension of each head; \( B \) denotes the deformable relative position bias \[55\].

Finally, \( F_{CA,m} \) is generated by the multi-head attention, which is then concatenated with \( F_{LA,m} \), i.e., the output of the local branch. This results in the output feature \( F_{out,m} \), which encodes the texture pattern for the input patch \( P_m \) via both local and global attention. \( F_{out,m} \) is then sent to the QENet as introduced in the next section.

### 4.3 Defocus-Aware Dynamic Quality Enhancement

Given the estimated defocus value (Section 4.1) and extracted texture pattern (Section 3.2) for the input patch, our DAQE approach can finally conduct the patch-wise dynamic quality enhancement via the proposed QENet, as presented in the following.

**Dynamic structure with multi-level enhancement.** The texture patterns at different clusters are more diverse compared with those at the same cluster, as revealed in Finding 3. It is therefore effective to enhance different clusters of patches in a “divide-and-conquer” manner, to finely restore the diverse texture patterns. To this end, we equip the QENet with a multi-level enhancement structure, which has \( N_{clu} \) levels of enhancement paths as shown in Figure 7(b). In this dynamic structure, the input feature can be enhanced through different levels of paths, which are determined dynamically according to their defocus values. Note that these paths are not independent; instead, they are connected progressively through the context adaptation (CA) subnets. Specifically, the feature in each path is adapted to the context information provided by the upper path, to take advantage of the similarity of texture patterns between these two neighboring clusters. The QENet is resource-efficient in the following two aspects. (1) Defocus-aware progressive enhancement. The input features of patches with lower defocus values are enhanced via more levels of paths, since they have inferior compression quality as observed in Finding 2. In the most sophisticated case, all levels of paths are traversed from top to bottom to pursue optimal enhancement performance. Conversely, those with higher defocus values are enhanced via only traversing the upper-level paths, such that the computational resources can be saved while remaining high quality. (2) Dynamic resolution of inference feature. The input feature is downsampled by different factors at different levels before enhancement. The enhanced image is finally upsampled to restore the resolution. This way, the upper-level enhancement are conducted over smaller inference features, thus consuming fewer computational resources.

**Quality enhancement at each level.** Here, we take the enhancement of the input feature \( F_{in,m} \) as an example, where \( m \) is the cluster index. Note that a cluster with a smaller \( m \) is with a higher center defocus value. Let \( n \) denote the level of the enhancement path. Then, the paths of the top-\( m \) levels are progressively traversed by \( F_{in,m} \), i.e., level \( n \) ranges from 1 to \( m \). Specifically, at the \( n \)-th level of enhancement, \( F_{in,m} \) is first downsampled with a factor of 2 for \( (N_{clu} - n) \) times. Then, the downsampled feature is encoded by residual blocks \( R_{enc,n} \) into \( F^{n}_{enc,m} \) as follows,

\[
F^{n}_{enc,m} = R_{enc,n}(Dw(\cdots(Dw(F_{in,m}))\cdots)), \tag{8}
\]

where \( Dw \) is a downsampling operator with a factor of 1/2. Subsequently, \( F^{n}_{enc,m} \) is further processed by the CA subnet, for adapting \( F^{n}_{enc,m} \) to a context feature and generating an adapted feature \( F^{n}_{ada,m} \). For the top-level path, \( F^{1}_{enc,m} \) serves as the context feature; for other paths, the output of the CA subnet at the upper path \( F^{n-1}_{ada,m} \) serves as the context feature. The CA subnet first convolves \( F^{n}_{enc,m} \) by a few residual blocks \( R_{in,n} \). It then convolves the context feature by residual blocks \( R_{\gamma,n} \) and \( R_{\beta,n} \) to obtain the adaption maps \( F^{n}_{\gamma,m} \) and \( F^{n}_{\beta,m} \), respectively. The adapted feature \( F^{n}_{ada,m} \) is produced by multiplying \( F^{n}_{\gamma,m} \) and then adding \( F^{n}_{\beta,m} \) to the convolved \( F^{n}_{enc,m} \). The above processes can be written as follows,

\[
F^{n}_{ada,m} = R_{in,n}(F^{n}_{enc,m}) \odot R_{\gamma,n}(F^{n-1}_{ada,m}) + R_{\beta,n}(F^{n-1}_{ada,m}), \tag{9}
\]

where \( F^{0}_{ada,m} \) refers to \( F^{1}_{enc,m} \). If the level index \( n \) meets the cluster index \( m \), \( F^{n}_{ada,m} \) is further sent to the decoder, i.e., a set of residual blocks \( R_{dec,n} \) and then upsampled to generate the enhanced patch \( P_{out,m} \) as follows,

\[
P_{out,m} = Up(\cdots(Up(R_{dec,n}(F^{n}_{ada,m})))\cdots), \tag{10}
\]

where Up is an upsampling operator with a factor of 2. Finally, we can obtain the output enhanced image \( I_{out} \) for the input compressed image \( I_{in} \) via spatially combing all enhanced patches of all clusters.

### 4.4 Loss Functions

We train our DAQE model in a supervised manner. Here, we discuss the loss functions for the supervision, which are composed of a quality enhancement loss and a defocus estimation loss.

**Quality enhancement loss.** Let \( L_{en} \) denotes the quality enhancement loss. Here, \( L_{en} \) is modeled by the Charbonnier loss function \[57\] between the enhanced patch \( P_{out} \) and the corresponding raw patch \( P_{r} \),

\[
L_{en} = \sqrt{\|P_{out} - \hat{P}_{r}\|^2 + \epsilon^2}, \tag{11}
\]

where \( \epsilon \) is a hyper-parameter for numerical stability. Then, the AGNet and QENet are trained in an end-to-end manner by minimizing \( L_{en} \).

**Defocus estimation loss.** We also take into account the defocus estimation loss \( L_{de} \) for training the DENet. Ideally, the ground truth defocus map for the compressed image is available for supervision. Unfortunately, it is impossible to obtain the ground-truth defocus map for an image. To solve this issue, we adopt the synthetic depth-of-field (SYNDOF) dataset \[21\] for training the DENet. The SYNDOF dataset...
contains 205 real defocused images $I_{\text{real}}$ without ground-truth defocus maps. It also contains 8,026 pairs of synthetic defocused images and defocus map $\{I_{\text{syn}}, M\}$. Note that $I_{\text{syn}}$ are synthesized by the thin-lens model \[58\] given $M$, as discussed in \[21\]. Then, we compress $I_{\text{real}}$ and $I_{\text{syn}}$ into real compressed images $I_{\text{c}}$ and synthetic compressed images $I_{\text{sc}}$, respectively. More details about image compression are discussed in Section 5.1. Finally, we estimate the defocus maps $M$ of $I_{\text{syn}}$ by the DENet, and obtain a set of $\{I_{\text{syn}}, M, M\}$ for supervision. Given the above training data of $I_{\text{c}}$ and $\{I_{\text{syn}}, M, M\}$, we define the defocus estimation loss $L_{\text{de}}$ as follows. First, we minimize the pixel-wise mean square error (MSE) between $M$ and $M$,

$$L_{\text{pix}} = \|M - M\|^2_2. \quad (12)$$

Then, we need to minimize the semantic distance between $M$ and $M$, measured by the feature-wise MSE,

$$L_{\text{feat}} = \|\phi(M) - \phi(M)\|^2_2, \quad (13)$$

where $\phi$ denotes the last convolution layer in the $l$-th block of a pre-trained VGG-19 model \[46\]. Next, we focus on reducing the domain gap between $I_{\text{c}}$ and $I_{\text{syn}}$ during the defocus estimation, through the following adversarial loss between their feature maps:

$$L_{\text{adv}} = \alpha \cdot \log(D(\psi(I_{\text{c}}))) + (1 - \alpha) \cdot \log(1 - D(\psi(I_{\text{syn}}))). \quad (14)$$

In the above equation, $I_{\text{c}}$ can be either $I_{\text{real}}$ or $I_{\text{syn}}$; $\psi$ denotes the last upsampling layer of the DENet; $D$ is a four-layer CNN-based discriminator; $\alpha$ is a label, and it equals to 0 when $I_{\text{c}} = I_{\text{real}}$ or equals to 1 when $I_{\text{c}} = I_{\text{syn}}$. Finally, the defocus estimation loss $L_{de}$ is modeled as follows:

$$L_{\text{de}} = L_{\text{pix}} + \lambda_{\text{feat}} \cdot L_{\text{feat}} + \lambda_{\text{adv}} \cdot L_{\text{adv}}, \quad (15)$$

where $\lambda_{\text{feat}}$ and $\lambda_{\text{adv}}$ are the weight factors. To obtain a converged discriminator, a discriminator loss $L_D = -L_{\text{adv}}$ is set to supervise the training of $D$. Given the above loss functions, we can train the DENet and the discriminator $D$ by alternately minimizing $L_{\text{de}}$ and $L_D$. 

5 Experiments

In this section, we present our experimental results to verify the performance of our proposed DAQE approach for quality enhancement on compressed images. Since BPG (HEVC-MSP) \[4\], \[5\] and JPEG \[2\] are two widely-used image compression codecs, our experiments mainly focus on enhancing the quality of both BPG and JPEG-compressed images.

5.1 Experimental Setup

In this section, we present details about the datasets, hyperparameters, training strategy, and testing procedure of our DAQE approach.

Datasets. Recent works have adopted some large-scale image datasets, such as BSDS500 \[63\] and ImageNet \[47\], for image denoising, segmentation, and other image tasks. However, the images from these datasets contain unknown artifacts, since they are collected under unknown conditions and compressed by unknown codecs and settings. 

To obtain “clean” images without significant artifacts, we adopt several high-quality image datasets for evaluation, as illustrated in Table 2. Specifically, we adopt 800 images of the DIV2K dataset \[22\] as the training set. Besides, we adopt all 25 images of the Kodak dataset \[59\], 100 images of the DIV2K dataset, 100 images of the Flickr2K dataset \[60\], and 100 images of the RAISE dataset \[61\] as the test set. We compress all images using the BPG \[7\] and JPEG codecs \[2\]. We adopt four compression settings for each codec, i.e., the quantization parameter (QP) is set to 27/32/37/42 in BPG and the quality factor (QF) is set to 20/30/40/50 in JPEG. Note that these settings are widely used for other quality enhancement works \[16\], \[20\], \[64\], \[65\], \[66\].

Hyper-parameters, training and testing. In our DAQE approach, $S$, $N_{\text{clu}}$, and $d$ are set to 128, 3, and 32, respectively. The number of attention heads is set to 3. All convolution operators are with a kernel size of 3, a stride of 1, and padding of 1. To cluster the input patches, we adopt the K-means clustering algorithm \[41\], \[42\] over the DIV2K training set. For the loss functions, we set $\epsilon, l, \lambda_{\text{feat}}$, and $\lambda_{\text{adv}}$ to $10^{-6}, 4, 10^{-4}$, and $10^{-3}$, respectively. During the training process, the Adam \[67\] optimizer is applied with an initial learning rate of $10^{-4}$; the cosine annealing schedule \[68\] is also applied to decrease the learning rate automatically. The training batch size is set to 64. A workstation with one CPU (Intel Xeon Platinum 8163 CPU @ 2.50GHz) and four GPUs (Tesla V100-SXM2-16GB) is used for both training and testing. We first train the DENet over the training set of SYNDOF. After the convergence of the DENet, we freeze the parameters of the DENet and train the subsequent AGNet and QENet jointly over the DIV2K training set until convergence.

5.2 Evaluation

In this section, we evaluate the performance of our DAQE approach in the quality enhancement of compressed images. We compare our approach with several widely-used approaches including AR-CNN \[11\], DCAD \[16\], DnCNN \[15\], CBDNet \[62\] and RBQE \[20\]. Among them, CBDNet and RBQE are originally used for blind restoration. For fair comparisons, we re-train them in a non-blind manner, i.e., train one model for each compression configuration. In addition, all compared approaches are re-trained over our training set.

Quantitative performance. To evaluate the efficacy of our DAQE approach, we measure PSNR and SSIM for different approaches on both BPG and JPEG-compressed images.

| Dataset | Max res. | Usage | Indices |
|---------|----------|-------|---------|
| DIV2K \[22\] | 2K | Training | 0001-0800 |
| Kodak \[59\] | 768x512 | Testing | 0001-0025 |
| DIV2K \[22\] | 2K | Testing | 0801-0900 |
| Flickr2K \[60\] | 2K | Testing | 2551-2650 |
| RAISE \[61\] | 4K | Testing | 8057-8156 |

TABLE 2

Datasets adopted in this paper. The maximal image resolution (res.), image usage, and image indices of these datasets are indicated.

2. Codes of all approaches are available at https://github.com/RyanXingQL/PowerVQE
images over four different datasets. Table 3 presents the results on BPG-compressed images. As shown in Table 3, the average PSNR of the DAQE approach on the DIV2K dataset is 32.69 dB at QP = 37, which is 1.24 dB higher than the BPG baseline, and 0.29 dB higher than that of the second-best approach. In addition, the average SSIM is 0.895, which is 0.018 higher than the BPG baseline, and 0.29 dB higher than that of the second-best approach. Similar results can be found for the other three datasets and other QF settings. In summary, the DAQE approach achieves state-of-the-art performance on all other three datasets and other QF settings. In summary, the DAQE approach outperforms the second-best CBDNet by 0.29 dB in PSNR with 8.67% higher FPS compared with the CBDNet.

### Rate-distortion performance

We further evaluate the rate-distortion performance of our DAQE approach in Figure 9 and Table 4. Figure 9 shows the rate-distortion curves of different approaches over the four datasets. As can be seen from this figure, the rate-distortion curves of our DAQE approach are higher than those of other approaches, indicating the better rate-distortion performance of our approach. Then, we quantify the rate-distortion performance by evaluating the reduction of Bjontegaard-rate (BD-rate) [69]. The results are presented in Table 4. As can be seen, for BPG-compressed images, the BD-rate reductions of our DAQE approach on the DIV2K dataset are averagely 24.05% and 22.14% with the distortion measured by PSNR and SSIM, respectively, while those of the second-best approach are only 19.38% and 16.45% on average. Similar results can be seen for the other three datasets and JPEG-compressed images. In summary, our DAQE approach significantly advances the state-of-the-art rate-distortion performance.

**Qualitative performance.** Figure 9 compares the visual results of our DAQE and the compared approaches. Specifically, the DAQE approach successfully restores the edge details of the door, motorbike, and window in Figure 9 (a)-(c), respectively. In contrast, these details cannot be well restored by other compared approaches. Besides, the DAQE approach suppresses the compression artifacts around these edges, while those artifacts can be hardly reduced by other compared approaches. To summarize, the DAQE approach outperforms the compared approaches qualitatively, especially in restoring details and suppressing compression artifacts.

**Efficiency.** We measure the efficiency of our DAQE and other compared approaches from two aspects: the time complexity in terms of the frames per second (FPS) and the space complexity in terms of the parameter number. As shown in Figure 10, the DAQE approach outperforms the second-best CBDNet by 0.29 dB in PSNR with 8.67% fewer parameters and 4.80% higher FPS. Some approaches, such as AR-CNN and DCAD, have fewer parameters and higher FPS compared with the DAQE approach and CBD-
There are some important components proposed in our DAQE network. First, a local attention module is designed in the AGNet to extract the texture pattern for each single input patch. In addition, the AGNet is equipped with a global attention module for extracting the texture pattern of the input patch by referring to all patches at the same cluster. Finally, the CA subnet is proposed to effectively connect each level of the QENet. To validate the effectiveness of these network components, we gradually ablate each component to generate three different networks.

### 5.3 Ablation Study

**Network components.** There are some important components proposed in our DAQE network. First, a local attention module is designed in the AGNet to extract the texture pattern for each single input patch. In addition, the AGNet is equipped with a global attention module for extracting the texture pattern of the input patch by referring to all patches at the same cluster. Finally, the CA subnet is proposed to effectively connect each level of the QENet. To validate the effectiveness of these network components, we gradually ablate each component to generate three different networks.
TABLE 6
Ablation results of our DAQE approach in terms of PSNR (dB).

| Component                     | DAQE  | (A) | (B) | (C) |
|-------------------------------|-------|-----|-----|-----|
| Local attention of AGNet      | ✔     | ✔   | ✔   | ✗   |
| Global attention of AGNet     | ✔     | ✔   | ✗   | ✗   |
| CA subnet of QENet            | ✔     | ✗   | ✗   | ✗   |
| PSNR (dB)                     | 32.69 | 32.59| 32.54| 32.51|

denoted by (A) to (C), as presented in Table 6. Then, we re-train and test all these networks over the DIV2K dataset compressed by BPG at QP = 37. As can be seen in Table 6, ablating the CA subnets degrades PSNR by 0.10 dB. The further ablation of the global attention and local attention lead to 0.05 and 0.03 dB degradation in PSNR, respectively. Thus, these network components bring a positive influence on the enhancement performance of the DAQE approach.

Defocus-based patch classification. During the process of defocus-based patch classification, we classify the patches into three different clusters by the DENet, to prepare for the subsequent dynamic quality enhancement. Figure 11 shows the statistics of the patches after the defocus estimation and clustering. As can be seen, the patches from different
clusters significantly differ in compression quality in terms of both the PSNR and SSIM values. Specifically, the median PSNR values of patches at three clusters are 39.21, 33.06, and 30.00 dB, respectively. The median SSIM values of patches at three clusters are 0.98, 0.94, and 0.92, respectively. The large quality gaps between patches at different clusters bring great benefits to cluster-specific quality enhancement. It is worth noting that the classification of compressed patches in our DAQE approach does not rely on their raw patches, making it practical in real-world applications.

**Defocus-based dynamic enhancement.** To evaluate the efficacy of the defocus-based dynamic enhancement of the DAQE approach, we design the following experiments. Specifically, instead of dynamic enhancement, we compulsively exit all patches at the first level of the QENet without considering their defocus values. The PSNR-FPS result is denoted by the blue square in Figure 12. Similarly, all patches exit at the second and the third levels of the QENet separately, generating two PSNR-FPS results also shown in Figure 12, respectively. Finally, each patch randomly exits with three different random seeds, generating three PSNR-FPS results as shown in Figure 12. As we can see from Figure 12, our defocus-based dynamic enhancement (denoted by the red star) achieves a superior trade-off between enhancement quality and speed, compared with other enhancement strategies.

\section{Conclusion}

In this paper, we have proposed the defocus-aware quality enhancement (DAQE) approach. Our DAQE approach considers the region-wise defocus difference of compressed images, thus differing from the traditional quality enhancement approaches in two aspects. (1) The DAQE approach
employs fewer computational resources to enhance the quality of significantly defocused regions, while more resources on enhancing the quality of other regions; (2) The DAQE approach learns to separately enhance diverse texture patterns for the regions with different defocus values, such that texture-wise one-on-one enhancement can be managed. To achieve these, the DAQE approach first estimates the defocus value for each image region with the proposed DENet. Next, patches are classified into different clusters according to their defocus values, and then sent to the AGNet and QENet to accomplish the cluster-specific texture extraction and dynamic quality enhancement. Finally, extensive experiments validated that our DAQE approach can significantly improve the quality of compressed images in a resource-efficient manner, superior to other state-of-the-art approaches.

There exist two research directions for future works. (1) Our work only takes PSNR and SSIM as the metrics for compression quality to be enhanced. The potential future work may explore other perceptual quality metrics to improve the QoE of compressed images, since the image defocus also correlates with the perceptual quality of compressed images. (2) Our work focuses on the quality enhancement of compressed images. The potential future work may extend the scope of defocus-aware approaches to other image enhancement and restoration tasks, e.g., image denoising and deblurring, because image defocus is inherent in the physics of image formation and can be utilized by more low-level vision tasks.

Fig. 10. Efficiency of our DAQE and compared approaches over the DIV2K test set compressed by BPG at QP = 37. The number of parameters is marked at the center of each circle. A bigger circle radius indicates a larger number of parameters.

Fig. 11. Statistics of PSNR and SSIM values for patches at different clusters. Patches are from the DIV2K test set compressed by BPG at QP = 37.

Fig. 12. PSNR-FPS performance of different enhancement strategies over the DIV2K test set compressed by BPG at QP = 37.

References

[1] D. Inc., “Data never sleeps 8:0. How much data is generated every minute?” [https://www.domo.com/learn/data-never-sleeps-8/2020], [Online; accessed 10-March-2021].
[2] G. Wallace, “The JPEG still picture compression standard,” IEEE Transactions on Consumer Electronics, vol. 38, no. 1, pp. xvii–xxxiv, 1992. [Online]. Available: https://doi.org/10.1109/30.125072
[3] M. Marcellin, M. Gormish, A. Bilgin, and M. Boliek, “An overview of JPEG-2000,” in Proceedings DCC 2000. Data Compression Conference. IEEE Comput. Soc, 2000. [Online]. Available: https://doi.org/10.1109%2Fdcc.2000.838192
[4] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, “Overview of the high efficiency video coding (HEVC) standard,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 22, no. 12, pp. 1649–1668, Dec 2012. [Online]. Available: https://doi.org/10.1109%2Ftcsvt.2012.2221191
[5] F. Bellard, “Better portable graphics (bpg),” [https://bellard.org/bpg/2018], [Online; accessed 10-March-2021].
[6] M.-Y. Shen and C.-C. Kuo, “Review of postprocessing techniques for compression artifact removal,” Journal of Visual Communication and Image Representation, vol. 9, no. 1, pp. 2–14, mar 1998. [Online]. Available: https://doi.org/10.1006%2Fjvci.1997.0378
[7] K. Seshadrinathan, R. Soundararajan, A. C. Bovik, and L. K. Cormack, “Study of subjective and objective quality assessment of video,” IEEE Transactions on Image Processing, vol. 19, no. 6, pp. 1427–1441, jun 2010. [Online]. Available: https://doi.org/10.1109%2Fticip.2010.2042111
[8] T. T. U. C. S. (ITU-T), “P10: Vocabulary for performance, quality of service and quality of experience,” https://www.itu.int/rec/R-REC-P10-Nov2017, [Online; accessed 30-July-2021].
[9] N. Salvaggio, Basic Photographic Materials and Processes. Routledge, apr 2013. [Online]. Available: https://doi.org/10.4324%2F9780080927664
Y. LeCun, Eds., 2015. [Online]. Available: http://arxiv.org/abs/1409.1556

[47] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A Large-Scale Hierarchical Image Database,” in CVPR09, 2009.

[48] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” in Lecture Notes in Computer Science. Springer International Publishing, 2015, pp. 234–241. [Online]. Available: https://doi.org/10.1007/978-3-319-25478-4_28

[49] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” in Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA, I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, and R. Garnett, Eds., 2017, pp. 5998–6008. [Online]. Available: https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html

[50] X. Wang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, “Image super-resolution using very deep residual channel attention networks,” in Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part VII, ser. Lecture Notes in Computer Science, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds., vol. 11211. Springer, 2018, pp. 294–310. [Online]. Available: http://openaccess.thecvf.com/content_CVPR_2018/paper/Wang_Recovering_Realistic_Texture_CVPR_2018_paper.html

[51] T. Park, M. Liu, T. Wang, and J. Zhu, “Semantic image synthesis with spatially-adaptive normalization,” in IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019. Computer Vision Foundation / IEEE Computer Society, 2018, pp. 606–615. [Online]. Available: http://openaccess.thecvf.com/content_cvpr_2018/html/Park_Semantic_Image_Synthesis_With_Spatially-Adaptive_Normalization_CVPR_2019_paper.html

[52] S. W. Zamir, A. Arora, S. H. Khan, M. Hayat, F. S. Khan, M. Yang, and L. Shao, “Multi-stage progressive image restoration,” in IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021. Computer Vision Foundation / IEEE, 2021, pp. 14821–14831. [Online]. Available: https://openaccess.thecvf.com/content/CVPR2021/html/Zamir_Multi-Stage_Progressive_image_Restoration_CVPR2021_paper.html

[53] Y. Zhao, X. Li, P. Song, S. Li, and G. Huang, “Vision transformer with deformable attention,” CoRR, vol. abs/2201.00520, 2022. [Online]. Available: http://arxiv.org/abs/2201.00520

[54] D. Hendrycks and K. Gimpel, “Gaussian error linear units (gelu),” arXiv preprint arXiv:1606.08415, 2016.

[55] P. Charbonnier, L. Blanc-Feraud, G. Aubert, and M. Barlaud, “A flexible framework for non-linear restoration via level set evolution,” in Proceedings of the 1994 International Conference on Image Processing, Austin, Texas, USA, November 13-16, 1994. IEEE Computer Society, 1994, pp. 168–172. [Online]. Available: https://doi.org/10.1109/ICIP.1994.413553

[56] D. Hendrycks and K. Gimpel, “Gaussian error linear units (gelu),” in Proceedings of the 8th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH’81, Dallas, Texas, USA, August 3-7, 1981, D. Green, T. Lucido, and H. Fuchs, Eds. ACM, 1981, pp. 297–305. [Online]. Available: http://doi.org/10.1145/800224.806818

[57] Kodak, “Kodak lossless true color image suite,” http://r0k.us/graphics/kodak/nov 1999.