Quality Adaptive Low-Rank Based JPEG Decoding with Applications

Xiao Shu   Xiaolin Wu

McMaster University

Abstract

Small compression noises, despite being transparent to human eyes, can adversely affect the results of many image restoration processes, if left unaccounted for. Especially, compression noises are highly detrimental to inverse operators of high-boosting (sharpening) nature, such as deblurring and superresolution against a convolution kernel. By incorporating the non-linear DCT quantization mechanism into the formulation for image restoration, we propose a new sparsity-based convex programming approach for joint compression noise removal and image restoration. Experimental results demonstrate significant performance gains of the new approach over existing image restoration methods.

1   Introduction

Image restoration is to improve the quality of acquired image data in preparation for higher level vision tasks. It remains a very active research area because the precision and success of many computer vision algorithms, such as registration, recognition, detection, classification, matting, retrieval, etc., depend on the quality of the input image. But much to our surprise, in the very large existing body of research literature on image restoration, very little study has been reported on how compression noises affect the performances of various image restoration processes, such as deconvolution, superresolution, etc. All published works of image restoration, except few papers explicitly on the topic of combating compression artifacts (a.k.a., soft decoding), assumed the input image data to be uncompressed or mathematically losslessly compressed. This long-time tradition is, unfortunately, an operational convenience in contrary to the real world settings. In most practical scenarios, particularly those of consumer applications, the input images are compressed in discrete cosine transform (DCT) domain with some loss of fidelity. For practical systems constrained by bandwidth and storage economy, lossy compression is inevitable because mathematically invertible image coding typically achieves only roughly 2:1 compression ratio,still leaving the image file size too large to handle.

Granted, after years of research, development and investment, international compression standards such as JPEG, H.264, HEVC, etc., can offer very high

reconstruction quality to the level of perceptual transparency; namely, naked eyes cannot discern any difference between the original and the decompressed images. But as demonstrated by this work, small compression noises, despite being transparent to human eyes, can adversely affect the results of many image restoration processes, if left unaccounted for. Especially, compression noises are highly detrimental to inverse operators of high-boosting (sharpening) nature, such as deblurring and superresolution against a convolution kernel.

The omission of compression noises in the design of image restoration algorithms is seemingly due to the fact that the compression noises are much more difficult to model than other degradation sources, e.g., motion blur and sensor noises. The compression-induced quality degradation is more pronounced for compound document images, which are characterized by the embedding of graphics arts or texts into an acquired photograph, as exemplified by Figure 1. The non-linearity of quantization operations in image compression systems makes quantization noises image dependent, far from being white and independent, as commonly assumed by most researchers in the field of image restoration.

The contributions of this paper are two folds. First, we analyze the nature of quantization noises in the DCT domain, in which most popular JPEG and H.264/HEVC compression standards operate. In particular, we find that the quantization errors of DCT coefficients exhibit complex behaviours after being mapped back into the spatial domain. These behaviours are highly sensitive to quantization precision, the amplitude and phase of the input image signal. Second, we manage to incorporate the non-linear DCT quantization mechanism into the inverse problem formulation for image restoration. Specifically, we propose a new sparsity-based convex programming approach for joint quantization noise removal and restoration, verify the efficacy of the proposed approach for the tasks of deblurring and super-resolving DCT-domain compressed images, and demonstrate significant performance gains of the new approach over existing deblurring and superresolution methods.

2   Quantization Error in DCT Domain

The process of capturing, storing and displaying a digital image is far from perfect; it often introduces objectionable errors, such as motion blur, lens distortion, moiré pattern, sensor noise, compression noise, etc., into the final reproduction of a scene. Some errors are independent to and statistically distinct from signal. For example, sensor noise can be modelled as random variables following an independent and identically distributed (i.i.d.) Gaussian distribution, while true signal has repetitive patterns hence sparse in some basis [13, 6]. By exploiting this statistical difference between signal and sensor noise, denoising techniques can effectively separate signal and noise in a given noisy observation [8]. Compression noises, on the other hand, are much more difficult to model than other degradation sources, e.g., motion blur and sensor noises. The non-linearity of quantization operations in image compression systems makes quantization

(a) Original image

(b) Enhanced by CLAHE [16]

(c) Enhanced by the proposed technique

Figure 1: JPEG compression artifacts become highly objectionable after image enhancement or image size magnification, or both. The second and third sub-figures from left are regions up-scaled by bi-cubic interpolation and A+ [18], respectively.

(a) Original (b) JPEG (QF = 75) (c) JPEG (QF = 50) (d) JPEG (QF = 25)
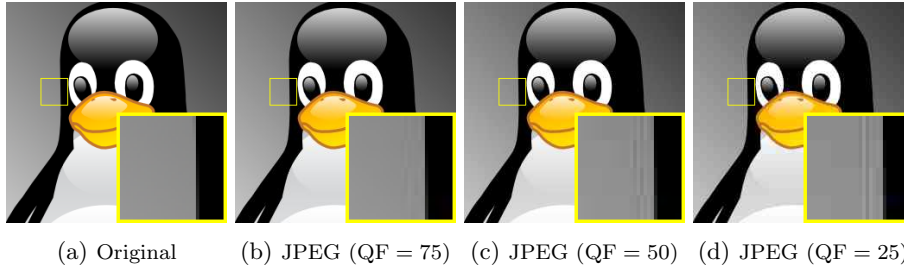
Figure 2: JPEG compression noise is corrected with signal.

noises image dependent, far from being white and independent.

In main stream DCT-based compression systems, the encoding of signal $\boldsymbol{x}$ is a three-step process. 1. The discrete cosine transform T is performed on signal $\boldsymbol{x}$; 2. the transformed signal $T(\boldsymbol{x})$ is subject to quantization Q; 3. the quantized version $(Q \circ T)(\boldsymbol{x})$ is coded by an entropy coder C, resulting the code stream $(C \circ Q \circ T)(\boldsymbol{x})$ for storage or transmission. The decoding process reverses the above three-step encoding process and generates the decompressed signal

$$\hat{\boldsymbol{x}} = (T^{-1} \circ Q^{-1} \circ C^{-1})((C \circ Q \circ T)(\boldsymbol{x})). \tag{1}$$

In this closed loop, the entropy decoder C^{-1} and the inverse transform T^{-1} are invertible operators, namely, $C^{-1} \circ C = I$, $T^{-1} \circ T = I$, but the dequantization operator Q^{-1} is not. The approximation error due to $Q^{-1} \circ Q \neq I$ is aggravated and complicated by the non-linearity of the quantization operation Q. In the interest of gaining compression performance, the quantizer Q inclines to demote or outright discard high-frequency DCT coefficients. Setting high frequency components of $\boldsymbol{x}$ to zero causes periodic ringing artifacts in the reconstructed signal $\hat{\boldsymbol{x}}$, which are easy to perceive as demonstrated in Figure 2. In this set of JPEG-decompressed images, the ringing artifacts not only accompany sharp edges in close proximity and they also agree with the image signal in orientation; in other words, the quantization noises are correlated with the image signal.

Unlike other noise mechanisms in image or video restoration, compression noises are not random in the sense that coding blocks of similar high-frequency contents tend to have similar ringing artifacts. As a result, a particular artifact pattern may occur repetitively in a pixel vicinity. Such signal-dependent noises may resist the treatment of sparsity-based denoising techniques, because the assumption that only the signal as self-similarity is no longer valid.

There is yet another complication in modeling and removing DCT quantization errors. That is, the same signal structure can, after through the loop of compression $C \circ Q \circ T$ and decompression $T^{-1} \circ Q^{-1} \circ C^{-1}$, exhibit much varied temporal or spatial patterns, with even immaterial changes in the phase or amplitude of the input signal, and in compression quality factor (QF). This high sensitivity and nonlinearity of error patterns are depicted graphically in Figure 3. In this example, the reconstructed versions of the same one dimensional (1D) unit pulse signal of minor linear-type alterations, such as shifting
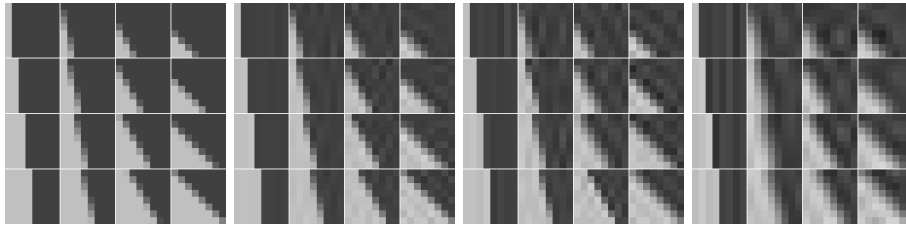
4
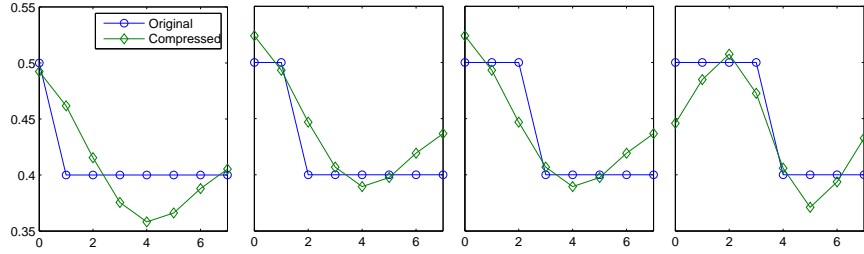
(a) Amplitude = 0.1, QF = 45.
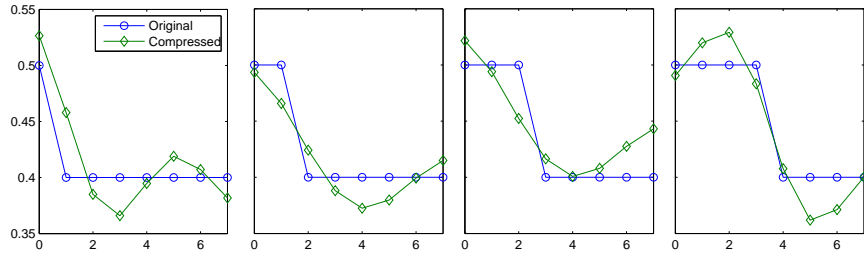
(b) Amplitude = 0.1, QF = 50.
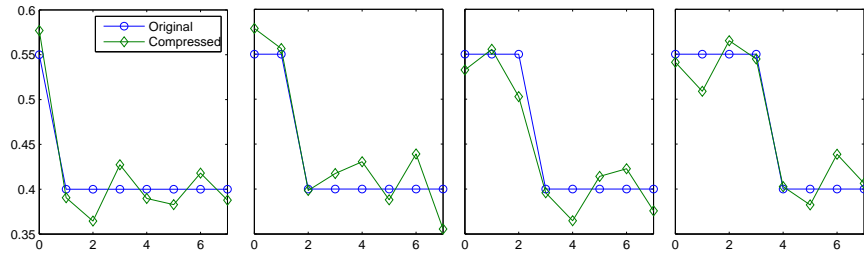
(c) Amplitude = 0.15, QF = 50.

Figure 3: DCT quantization noise appears drastically different with small changes in QF, signal phase or amplitude.

(a) Original   (b) JPEG (QF = 75)   (c) JPEG (QF = 50)   (d) JPEG (QF = 25)

Figure 4: Besides QF, JPEG compression noise is sensitive to the angle and phase of the signal.

and scaling, behave drastically differently. In two dimensional (2D) image, there are more factors affecting quantization noise. Figure 4 shows the quantization effects on image blocks of the same sharp edge but different phases and angles. In each case, there are visible false lines parallel to the edge, however, the position, strength and sign of the compression noise vary with a small change in angel or phase. Since there are so many factors affecting the quantization error, using learning based techniques to build a map from noisy observation to true signal for each scenario is impractical.

3 DCT Quantization Error Model

DCT based lossy compression techniques realize data volume reduction by trading off the accuracy of the DCT-domain representation of the input signal through quantization. By the definition of DCT, the k-th DCT coefficient of 1D signal $x_0, \ldots, x_{N-1}$ is,

$$X_k = \sum_{n=0}^{N-1} x_n \cos\left[\frac{\pi}{N}\left(n+\frac{1}{2}\right)k\right]. \tag{2}$$

After quantization, the true value of X_k is commonly estimated as,

$$\hat{X}_k = \lfloor X_k/q_k + 0.5 \rfloor \cdot q_k, \tag{3}$$

where q_k is quantization interval for the k-th DCT coefficient. In general, q_k is set to decrease with k, due to the fact that most energy of a signal is commonly concentrated in low frequency components.

Using Fourier transform, DCT can be approximated in continuous domain as follows,

$$\begin{aligned}
X_k &= N\sum_{n=0}^{N-1} \frac{1}{N} x_n \cos\left(2\pi \cdot \frac{n+\frac{1}{2}}{N} \cdot \frac{k}{2}\right) \\
&\approx N\int_0^1 f(t)\cos\left(2\pi t\frac{k}{2}\right)\mathrm{d}t \\
&= \mathrm{Re}\left[\frac{N}{2}\int_{-\infty}^{\infty} f(t)e^{-2\pi t\frac{k}{2}}\,\mathrm{d}t\right] \\
&= \mathrm{Re}\left[\frac{N}{2}F\left(\frac{k}{2}\right)\right],
\end{aligned} \tag{4}$$

where f is an integrable function such that

$$\begin{cases} f(\frac{n+\frac{1}{2}}{N}) = x_n, \\ f(t) = f(-t), \\ f(t) = 0, \quad t > 1, \end{cases} \tag{5}$$

and F is the Fourier transform of f. By this equation, if sequence $x_0, \ldots, x_{N-1}$ consists of equally spaced samples of function f and f satisfies Eq. (5), then a DCT coefficient of the sequence is a sample of f in frequency domain.
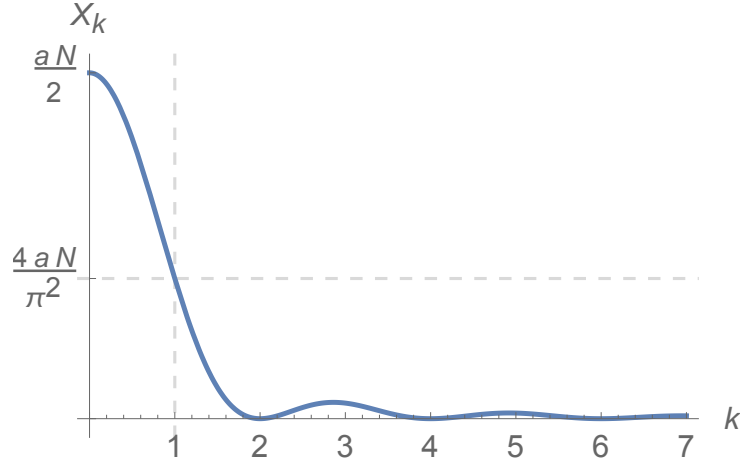
Figure 5: DCT coefficients of a decreasing linear sequence.

3.1 Quantization Effects on Linear Signal

Suppose input signal $\boldsymbol{x}_r = \{x_0, \ldots, x_{N-1}\}$ is a decreasing linear sequence, in which the n-th element is,

$$x_n = a \cdot \frac{n + \frac{1}{2}}{N}, \quad 0 \leq n \leq N - 1. \tag{6}$$

Then triangular function $a \cdot \text{tri}(t)$, where,

$$\text{tri}(t) = \begin{cases} 1 - |t| & \text{if } |t| < 1 \\ 0 & \text{otherwise,} \end{cases} \tag{7}$$

satisfies the conditions in Eq. (5), thus by Eq. (4), the k-th DCT coefficient of the sequence can be approximated as,

$$
\begin{aligned}
X_k &\approx \text{Re}\left[\frac{N}{2} \int_{-\infty}^{\infty} a \cdot \text{tri}(t) e^{-i2\pi t \frac{k}{2}} \, \mathrm{d}t \right] \\
&= \frac{aN}{2} \cdot \text{sinc}^2\left(\frac{k}{2}\right),
\end{aligned} \tag{8}
$$

where $\text{sinc}(\cdot)$ is the normalized sinc function defined as,

$$\text{sinc}(x) = \frac{\sin(\pi x)}{\pi x}. \tag{9}$$

This linear input signal $\boldsymbol{x}_r$ is easy to model in temporal domain; its second order derivative is zero everywhere hence sparse. It is comparable to a simple gradient ramp in 2D digital image. However, as visualized in Fig. 5, this signal

7

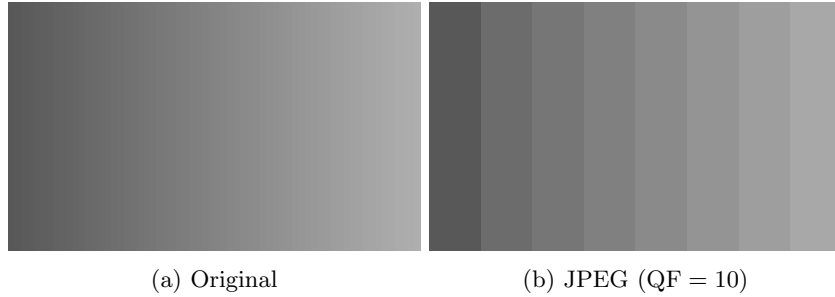<div align="center">(a) Original   (b) JPEG (QF = 10)</div>

Figure 6: Quantization removes small AC components causing perceivable blocking artifacts in a simple gradient ramp image.

is not sparse in DCT domain. By the approximation in Eq. (8), the k-th DCT coefficient X_k is only zero for even positive integer k, thus, more than half ($\lfloor N/2 \rfloor + 1$ out of N) of the coefficients in DCT domain are non-zero. To effectively compress this signal, some of the non-zero DCT coefficients need to be quantized to zero. Suppose for some odd positive integer k_0, the k_0-th quantized DCT coefficient is zero, then

$$X_{k_0} < \frac{q_{k_0}}{2} \iff \frac{aN}{2} \cdot \mathrm{sinc}^2 \left(\frac{k_0}{2} \right) < \frac{q_{k_0}}{2}$$

$$\iff aN \cdot \frac{4}{\pi^2 k_0^2} < q_{k_0}$$

$$\iff a < \frac{\pi^2 k_0^2 q_{k_0}}{4N}. \tag{10}$$

Since non-zero DCT coefficient X_k decreases with k while in general, quantization interval q_k increases with k, all the quantized DCT coefficients after the k_0-th one are zero as well. For example, if we use JPEG with QF = 25 to compress a horizontal gradient ramp, then by Eq. (10), quantized DCT coefficient X_1 is zero when a is less than about 4.8. In this case, each coding block becomes uniform after compression as its AC components in DCT domain are all zeros. As shown in Fig. 6, it is not sufficient to reconstruct the gradient ramp accurately in each block with only the DC component X_0. More importantly, due to Mach bands illusion, the discontinuity around coding block boundaries is highly perceivable to human, greatly deteriorating the perceptual quality of the compressed image.

3.2 Quantization Effects on Piecewise Constant Signal

Similar to linear signal, piecewise constant signal is another case which is simple to model in temporal domain but complex in DCT domain. For instance, let

input signal $\boldsymbol{x}_s = \{x_0, \ldots, x_{N-1}\}$ be a sequence of two steps, i.e.,

$$[\underbrace{a, a, \ldots, a}_{m}, \underbrace{0, 0, \ldots, 0}_{N-m}], \tag{11}$$

where $a > 0$ and $0 \leq m \leq N$. This sequence is a discrete version of rectangular function $f_s(t)$, where,

$$
\begin{aligned}
f_s(t) &= a \cdot \mathrm{rect}\left(\frac{t}{2r}\right) \\
&= a \cdot \begin{cases} 0 & \text{if } |t| > r \\ \frac{1}{2} & \text{if } |t| = r \\ 1 & \text{if } |t| < r, \end{cases}
\end{aligned} \tag{12}
$$

and $r = m/N$. As $f_s(t)$ satisfies Eq. (5), the DCT of sequence $\boldsymbol{x}_s$ can be approximated as follows by Eq. (4),

$$
\begin{aligned}
X_k &\approx \mathrm{Re}\left[\frac{N}{2}\int_{-\infty}^{\infty} a \cdot \mathrm{rect}\left(\frac{t}{2r}\right) e^{-i2\pi t \frac{k}{2}}\,\mathrm{d}t\right] \\
&= \frac{N}{2} \cdot 2ar \cdot \mathrm{sinc}\left(2r \cdot \frac{k}{2}\right) \\
&= arN \cdot \mathrm{sinc}(rk).
\end{aligned} \tag{13}
$$

As $\mathrm{sinc}(rk)$ decreases with frequency k in general, if quantization intervals are large enough, quantization effects can be approximated by cutting off high frequency components. Suppose only the first b DCT coefficients are preserved after quantization, then the restored sequence is

$$
\begin{aligned}
\hat{x}_t &= \int_{-b}^{b} \frac{2}{N} X_k\, e^{i2\pi t \frac{k}{2}}\,\mathrm{d}\frac{k}{2} \\
&= a\int_{-b}^{b} r\,\mathrm{sinc}(rk)\, e^{i2\pi \frac{t}{2}k}\,\mathrm{d}k \\
&= a \cdot [\mathrm{Si}(br - bt) + \mathrm{Si}(br + bt)],
\end{aligned} \tag{14}
$$

where function $\mathrm{Si}(z)$ is sine integral defined as

$$\mathrm{Si}(z) = \int_0^z \mathrm{sinc}(t)\,\mathrm{d}t. \tag{15}$$

By aligning the sequence $\{\hat{x}_t \mid 0 \leq t \leq 1\}$ to the location of the edge, we get the following sequence such that y_0 is the transition of two steps,

$$\hat{y}_t = \hat{x}_{t+r} = a[\mathrm{Si}(-bt) + \mathrm{Si}(2br + bt)]. \tag{16}$$

As shown in Fig. 7, quantization noise in $\hat{y}_t$ has a relatively fixed pattern regardless of the phase. Therefore, if we align the signals by their phases, the
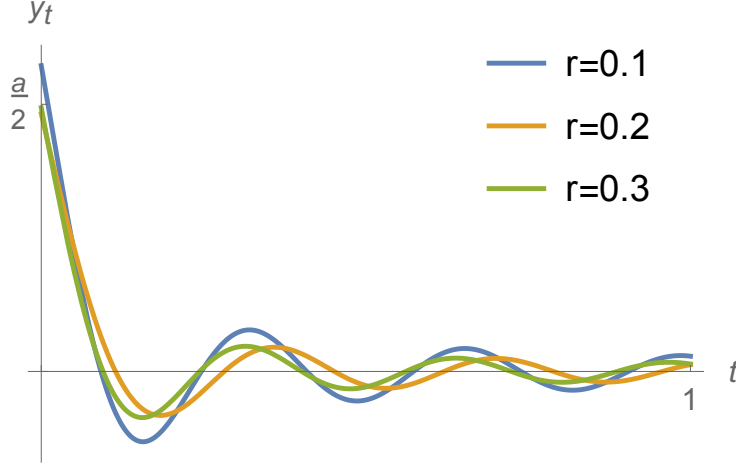
9

Figure 7: A sharp edge causes similar quantization artifacts regardless of the phase r.

noises become aligned as well. This correlation between signal and quantization noise makes them much more difficult to distinguish.

If sequence $\boldsymbol{x}_s$ is smoothed with a Gaussian kernel resulting sequence $\boldsymbol{y}_s$, then $\boldsymbol{y}_s$ is a discrete version of $f_g(t) = (f_s * g)(t)$, where,

$$g(t) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{t^2}{2\sigma^2}}. \tag{17}$$

By convolution theorem, the Fourier transform of function $f_g(t)$ is,

$$F_g\left(\frac{k}{2}\right) = F_s\left(\frac{k}{2}\right) \cdot G\left(\frac{k}{2}\right)$$

$$= 2ar \cdot \operatorname{sinc}(rk) \cdot e^{-2\pi^2\sigma^2 k^2/2^2} \tag{18}$$

Thus the k-th DCT coefficient of the blurred sequence $\boldsymbol{y}_s$ is approximately equal to,

$$Y_k \approx arN \cdot \operatorname{sinc}(rk) \cdot e^{-\pi^2\sigma^2 k^2/2} \tag{19}$$

For a given frequency k, the absolute quantization error is

$$\epsilon_k = |v_k \cdot q_k - Y_k|. \tag{20}$$

where $v_k = \lfloor Y_k/q_k + 0.5 \rfloor$. If the absolute quantization error ϵ_k of DCT coefficient Y_k is sufficiently small, say less than a constant C_ϵ, it has little impact on the quality of the compressed image; if ϵ_k is large, but the relative error $\epsilon_k/|Y_k|$ is small, it still contributes little to the artifacts of the compressed image, as in this case, the quantized DCT coefficient is strong enough to hide the error
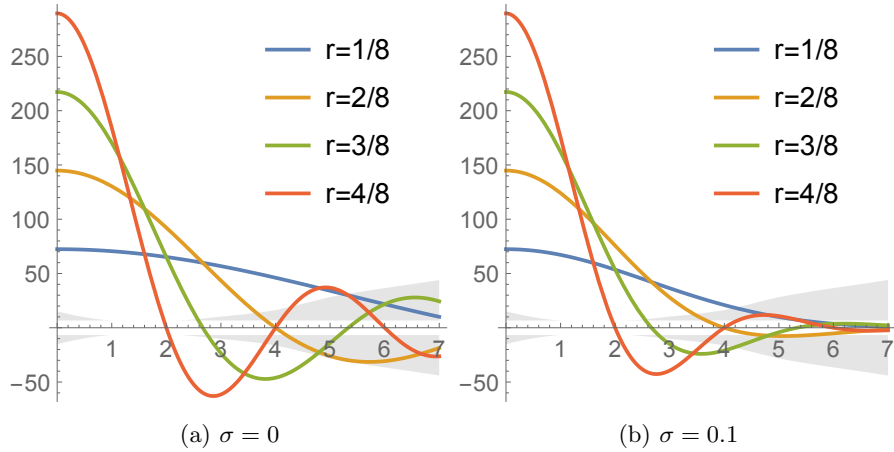
10

(a) $\sigma = 0$ (b) $\sigma = 0.1$

Figure 8: DCT coefficient Y_k as a function of frequency k.

perceptually. Suppose that to hide the quantization artifacts from frequency k, the relative error $\epsilon_k/|Y_k|$ must be less than $1/3$, i.e.,

$$\frac{e_k}{|Y_k|} = \left| \frac{v_k \cdot q_k - Y_k}{Y_k} \right| < \frac{1}{3}. \tag{21}$$

This inequality is true if and only if $|Y_k| \geq 3q_k/4$. Thus, when,

$$C_\epsilon \leq |Y_k| \leq \frac{3}{4}q_k, \tag{22}$$

quantization of DCT coefficient Y_k results both large absolute error and relative error.

Plotted in Fig. 8 is /glsdct coefficient Y_k as a function of k. Each curve represents a signal with a different phase r, and amplitude of the first step is $a = 50$; and the quantization intervals are based on the quantization matrix of JPEG with QF $= 50$. Regions where a coefficient can cause large absolute error and relative quantization error are marked as gray. As demonstrated in Fig. 8a, the strength and sign of a /glsdct coefficient and its quantization error depend on various factors, e.g., phase r, smoothness σ and amplitude a. In Fig. 8b, the sequence is smoothed by a Gaussian kernel with variance σ^2. As a result, the quantization errors of high frequency coefficients are small (not in the gray regions) compare to the previous case, as those coefficients are close to zero.

4 Enhancement Model

Recent nonlocal self-similarity (NNS) based image denoising techniques, such as BM3D [5], SAIST [7] and WNNM [8], have demonstrated their great strength in reconstructing the original image $\boldsymbol{x}$ from an observation $\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{n}$ contaminated

by additive white Gaussian noise $\boldsymbol{n}$. NNS refers to the fact that there are many repeated local patterns across a natural image, and those nonlocal similar patches to a given patch can help much the reconstruction of it [3]. For a local patch $\boldsymbol{y}_i$ of size m in image $\boldsymbol{y}$, we can stack M of its similar patches $\boldsymbol{y}_{i,j} = \boldsymbol{R}_{i,j} y$ across the image together into a matrix $\boldsymbol{Y}_i \in \Re^{m \times M}$, where $\boldsymbol{R}_{i,j}$ is a matrix extracting the j-th similar patch of the local patch at location i for $1 \leq i \leq N, 1 \leq j \leq M$. Then solving the following nuclear norm minimization (NNM) problem yields a matrix $\hat{\boldsymbol{X}}_i$ consisting of noise reduced patches,

$$\hat{\boldsymbol{X}}_i = \underset{\boldsymbol{X}_i}{\operatorname{argmin}} \|\boldsymbol{Y}_i - \boldsymbol{X}_i\|_F^2 + \lambda \|\boldsymbol{X}_i\|_* \tag{23}$$

Although this problem in non-convex, it is tractable by an efficient singular value thresholding (SVT) algorithm [3]. The whole reconstructed image can be then estimated by aggregating all the denoised patches as,

$$\hat{\boldsymbol{x}} = \left(\sum_{i=1}^{N} \sum_{j=1}^{M} \boldsymbol{R}_{i,j}^{\mathsf{T}} \boldsymbol{R}_{i,j} \right)^{-1} \sum_{i=1}^{N} \sum_{j=1}^{M} \boldsymbol{R}_{i,j}^{\mathsf{T}} \hat{\boldsymbol{x}}_{i,j} \tag{24}$$

Following this idea, we can formulate the restoration of DCT-domain compressed image problem as a constrained nuclear norm minimization problem,

$$\begin{aligned}
\hat{\boldsymbol{x}} = \underset{\boldsymbol{x}}{\operatorname{argmin}} \quad & \sum_{i=1}^{N} \|\boldsymbol{X}_i\|_* \\
\text{s.t.} \quad & |\boldsymbol{QT}\tilde{\boldsymbol{R}}_i \boldsymbol{Hx} - \boldsymbol{\gamma}_i| \leq 0.5, \\
& i = 1 \dots n
\end{aligned} \tag{25}$$

where $\boldsymbol{H}$ is a matrix modelling the degradation of image quality caused by various image capturing conditions, $\tilde{\boldsymbol{R}}_i$ is a matrix extracting the coding block at location i, $\boldsymbol{T}$ is the DCT transform matrix, $\boldsymbol{Q}$ is a diagonal matrix storing quantization table and vector $\boldsymbol{\gamma}_i$ is the DCT coefficient of the block at location i.

Using hard-decoding technique, each block $\boldsymbol{y}_i$ of $\boldsymbol{y}$ is obtained by inverse DCT transform from coefficient vector $\boldsymbol{\gamma}_i$ as follow,

$$\boldsymbol{y}_i = \tilde{\boldsymbol{R}}_i \boldsymbol{y} = (\boldsymbol{QT})^{-1} \boldsymbol{\gamma}_i, \tag{26}$$

and the observed image $\boldsymbol{y}$ is a degraded version of image $\boldsymbol{Hx}$ contaminated mainly by DCT-domain quantization noise. Existing sparsity based denoising techniques designed for reducing additive white Gaussian noise generally use $\|\hat{\boldsymbol{x}} - \boldsymbol{y}\|_2$ as the fidelity term in their optimization frameworks and leave $\hat{\boldsymbol{x}}$ unconstrained. In the case of DCT-domain quantization noise, we have more information about the noise: the true value of each DCT coefficient before scalar quantization lies in a known interval,

$$\boldsymbol{Q}^{-1}(\boldsymbol{\gamma}_i - 0.5) \leq \boldsymbol{T}\tilde{\boldsymbol{R}}_i \boldsymbol{Hx} < \boldsymbol{Q}^{-1}(\boldsymbol{\gamma}_i + 0.5) \tag{27}$$

This constrain confines the solution space of the optimization problem in Eq. (25), preventing the sparsity objective function from over-smoothing the output image.

To solve the problem in Eq. (25), we split it into two parts. The first part is to find a sparse estimation $\hat{z}$ of the original image from a given noisy version $\hat{x}$, i.e., for each patch group of image $\hat{z}$,

$$\hat{Z}_i = \operatorname*{argmin}_{Z_i} \|\hat{X}_i - Z_i\|_F^2 + \lambda \|Z_i\|_* \tag{28}$$

The noisy version $\hat{x}$ of the original image can be estimated directly using $\hat{x} = H^{-1}y$. Here we use H^{-1} to represent an inverse operator of H rather than matrix inverse. Although many types of image degradation can be modelled by a simple product of a degradation matrix H and the original image x, the inverse problem is often iso-posed and requires complex non-linear algorithm to find a good solution. Since the observed image y contains compression noise, if operator H^{-1} exhibits high-boosting property, which is often the case for unsharp and edge enhancement operators, H^{-1} could amplify the noise and make an inaccurate estimation of the original image. Using sparsity prior, the boosted noise can be greatly alleviated by the optimization problem in Eq. (28), resulting a better estimation of the original image in vector $\hat{z}$.

The second part of the problem is to impose the DCT-domain constraint in Eq. (25) on the noise reduced estimation $\hat{z}$ from the first part using the following optimization problem,

$$
\begin{aligned}
\hat{x}' = \operatorname*{argmin}_{x} \quad & \|x - \hat{z}\|_2 \\
\text{s.t.} \quad & |QT\tilde{R}_i Hx - \gamma_i| \le 0.5, \\
& i = 1 \ldots n
\end{aligned}
\tag{29}
$$

This is a convex problem solvable by off-the-shelf convex optimization problem solvers. However, if the input image is large, a general purpose solver is too time-consuming for this problem. Instead, we can solve a similar but much simpler problem as follows,

$$
\begin{aligned}
\hat{x}' = \operatorname*{argmin}_{x} \quad & \sum_{i=1}^{n} \|\tilde{R}_i H(x - \hat{z})\|_2 \\
\text{s.t.} \quad & |QT\tilde{R}_i Hx - \gamma_i| \le 0.5, \\
& i = 1 \ldots n
\end{aligned}
\tag{30}
$$

Compared with the original problem, the only difference of the reduced problem is that the new problem measures the norm of the error in degraded image domain rather than original image domain. Since the DCT tranform matrix T is unitary,

$$
\begin{aligned}
\|\tilde{R}_i H(x - \hat{z})\|_2 &= \|T\tilde{R}_i H(x - \hat{z})\|_2 \\
&= \|Ty_i - T\tilde{R}_i H\hat{z}\|_2,
\end{aligned}
\tag{31}
$$

where coding block $\boldsymbol{y}_i = \tilde{\boldsymbol{R}}_i \boldsymbol{H} \boldsymbol{x}$. On the other hand, the DCT coefficient constraint in Eq. (27) is applied on each element of vector $\boldsymbol{T} \boldsymbol{y}_i$, thus, the optimization problem has a closed-form solution,

$$\hat{\boldsymbol{y}}_i = \mathcal{C}_{0.5}(\tilde{\boldsymbol{R}}_i \boldsymbol{H} \hat{\boldsymbol{z}}, \boldsymbol{y}_i), \tag{32}$$

where, the DCT-domain clipping operator $\mathcal{C}_\beta(\cdot, \cdot)$ is defined as

$$\begin{aligned} \mathcal{C}_\beta(\boldsymbol{\alpha}, \boldsymbol{\rho}) = (\boldsymbol{Q}\boldsymbol{T})^{-1} \min(\max(\boldsymbol{Q}\boldsymbol{T}\boldsymbol{\alpha}, \boldsymbol{Q}\boldsymbol{T}\boldsymbol{\rho} - \beta) \\ , \boldsymbol{Q}\boldsymbol{T}\boldsymbol{\rho} + \beta). \end{aligned} \tag{33}$$

Aggregating these DCT blocks together, we get $\hat{\boldsymbol{y}}$, an estimation in degraded image domain with reduced compression noise, from which an approximate solution $\hat{\boldsymbol{x}}' = \boldsymbol{H}^{-1} \hat{\boldsymbol{y}}$ of the problem in Eq. (30) can be easily found. Compared with image $\hat{\boldsymbol{x}}$, the initial inverse of the observed image $\boldsymbol{y}$, $\hat{\boldsymbol{x}}'$ has lower level of compression noise because of sparsity prior in Eq. (28) and still satisfies the DCT domain constraints due to Eq. (30).

5   Algorithm

Algorithm 1 Image restoration from compressed image

Input: Compressed image $\boldsymbol{y}$, contrast degradation matrix $\boldsymbol{H}$

1: Estimate compression RMSE ε of $\boldsymbol{y}$
2: Estimate threshold λ using ε ▷ Eq. (43)
3: $\beta = 0.2$
4: $\boldsymbol{x}^{(0)} = \boldsymbol{H}^{-1} \boldsymbol{y}$
5: **for** $k = 1$ to K **do**
6:     **for each** patch $\boldsymbol{x}_i$ in $\boldsymbol{x}^{(k-1)}$ **do**
7:         Find similar patch group $\boldsymbol{X}_i$
8:         $[\boldsymbol{U}, \boldsymbol{\Sigma}, \boldsymbol{V}] = \mathrm{SVD}(\boldsymbol{X}_i)$
9:         $\boldsymbol{Z}_i = \boldsymbol{U} \mathcal{T}_\lambda(\boldsymbol{\Sigma}) \boldsymbol{V}^{\mathsf{T}}$ ▷ Eq. (38)
10:    **end for**
11:    Aggregate $\boldsymbol{Z}_i$ to form image $\boldsymbol{z}^{(k)}$ ▷ Eq. (24)
12:    **if** $k = K$ **then**
13:       $\beta = 0.5$
14:    **end if**
15:    Clip $\boldsymbol{z}^{(k)}$ using threshold β to get $\boldsymbol{y}^{(k)}$ ▷ Eq. (32)
16:    $\boldsymbol{x}^{(k)} = \boldsymbol{H}^{-1} \boldsymbol{y}^{(k)}$
17:    $\lambda = \lambda/2$
18: **end for**

Output: Restored image $\boldsymbol{x}^{(K)}$

Based on the restoration model discussed in the previous section, our purposed algorithm can be implemented as an iterative process alternatively finding
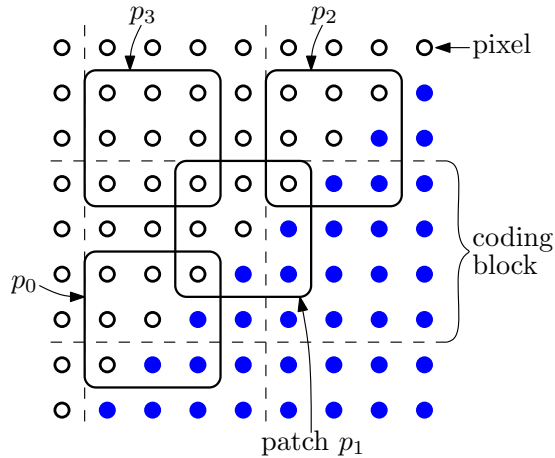
Figure 9: Avoid collecting patches of the same phase.

a reconstructed original image and a compression noise reduced observation image as in Algorithm 1. In this section, we address some of the technical issues in the implementation of the algorithm.

5.1 Nonlocal Self-Similarity

NNS based techniques have achieved the state-of-the-art results in removing Gaussian noise [8]. The NNS prior assumes that noise is independent to signal, hence by comparing a group of similar patches, noise can be isolated from signal. However, as we argued in previous sections, compression noise is not random but correlated with the signal; similar patches have similar compression noise, especially when they also have the same relative position to DCT coding blocks. Moreover, patches with matched artifacts can be easily mistaken as being similar using square error metric. Thus, collecting similar patches without taking their contents or positions into consideration inevitably puts multiple instances of the same quantization artifacts into a sample patch group; consequently, such reoccurring noises cannot be separated from the true signal by the NNS prior alone.

For example, as shown in Fig. 9, patches p_0 and p_2 are both located on a 45° high-contrast edge, and their positions relative to coding blocks are the same, hence they have matching ringing artifacts caused by the quantization of the edge in DCT domain. In contrast, patch p_1 on the same edge also suffers from ringing artifacts but with a different pattern than those of p_0 and p_1 as a result of being aligned differently to coding blocks than the other two. Due to its distinct noise patterns, patch p_1 is ranked lower in terms of the similarity to p_0, however, it is a better candidate for the sample patch group in combating reoccurring artifacts. Therefore, when compiling the sample patch group for patch p_0, other patches of the same position in relative to coding blocks, like p_2,

15

should be avoided if p_0 is around a high-contrast edge. A special case is that, when the edge is horizontal or vertical, patches in the same row or column are potentially distorted by the same artifacts, thus their similarity rating must be reduced accordingly as well.

In addition to considering patch positions in choosing similar patches, the measurement of patch similarity should be carefully designed to decouple noise from signal. As discussed previously, compression noise in input image can mislead the selection of similar patches. Thus, instead of comparing two patches directly by their squared error, a denoised version of a patch can be used to measure similarity. A simple low-pass filtering can generate a good enough denoised image effectively reducing the impact of compression noise to the measurement of similarity. This technique is only used in the first iteration of a denosing algorithm when the strength of noise is high. In the later iterations, input image becomes less subject to compression noise and it is not necessary to clean up the input image for a more robust measurement of similarity.

The above discussed techniques are designed to deal with patches with ringing artifacts around strong edges. Patches in smooth areas are generally free of ringing artifacts since their high frequency coefficients are near zero and the corresponding quantization errors are negligible. However, these patches are not immune to blocking artifacts. Due to the lack of other textures, the boundaries of coding blocks are actually more discernible perceptually in those smooth areas as demonstrated in Fig. 6. To prevent these blocking artifacts being matched as similar patch features causing reoccurring artifacts in sample patch group, the same strategy of choosing only unaligned patches as previous case can be employed. For example, in Fig. 9, patch p_3 is in a smooth area located across two coding blocks; any patch in the same row as p_3 is likely to have the identical blocking artifacts, hence it should not be considered in the patch group of p_3. Moreover, since natural images are smooth in general, patches in a small windows of smooth area are similar to each other. Furthermore, since natural images are smooth in general, patches in a small windows of smooth area are similar to each other. Therefore, for a patch from a smooth area, patches in close proximity are sufficient to build a good sample patch group. If the search window is small enough, there are few patches perfectly aligned with the given patch, hence reducing the risk of collecting too many patches with repeated artifacts. In practice, we set the search windows to 60×60 for normal patch and shrink the window to 10×10 when the variance of the given patch is less than 3.

5.2 Singular Value Thresholding

Ideally, finding a low-rank reconstruction of patch group matrix $\boldsymbol{Y}_i$ should be formulated as an ℓ_0-norm minimization problem as follows,

$$\hat{\boldsymbol{X}}_i = \operatorname*{argmin}_{\boldsymbol{X}_i} \|\boldsymbol{Y}_i - \boldsymbol{X}_i\|_F^2 + \lambda \|\boldsymbol{X}_i\|_0. \tag{34}$$

Since this problem is NP-hard [3], in practice, we approximate it with a nuclear norm minimization problem as in Eq. (23), which has an efficient closed-form solution,

$$\hat{\boldsymbol{X}}_i = \boldsymbol{U}\mathcal{D}_\lambda(\boldsymbol{\Sigma})\boldsymbol{V}^\mathsf{T}, \tag{35}$$

where $\boldsymbol{U}, \boldsymbol{\Sigma}, \boldsymbol{V}$ represent the the singular value decomposition (SVD) of $\boldsymbol{Y}_i$ and $\mathcal{D}_\lambda(\cdot)$ is a soft-thresholding operator,

$$\mathcal{D}_\lambda(\boldsymbol{\Sigma})_{jj} = \begin{cases} \boldsymbol{\Sigma}_{jj} - \lambda & \boldsymbol{\Sigma}_{jj} > \lambda, \\ 0 & \text{otherwise}, \end{cases} \tag{36}$$

or simply $\mathcal{D}_\lambda(\boldsymbol{\Sigma})_{jj} = \max(\boldsymbol{\Sigma}_{jj} - \lambda, 0)$.

Although this is a reasonable approximation employed by many applications [3, 21, 7], it still has some weaknesses. One of its problems is that, in addition to having a lower rank, the Frobenius norm of the optimal solution $\hat{\boldsymbol{X}}_i$ also decreases with larger threshold λ, since,

$$\|\hat{\boldsymbol{X}}_i\|_F^2 = \|\boldsymbol{U}\mathcal{D}_\lambda(\boldsymbol{\Sigma})\boldsymbol{V}^\mathsf{T}\|_F^2 = \sum_{j=1}^M \mathcal{D}_\lambda(\boldsymbol{\Sigma})_{jj}^2 \tag{37}$$

and $\mathcal{D}_\lambda(\boldsymbol{\Sigma})_{jj}$ is a decreasing function to λ. In the context of image denoising, when we try to increase the strength of the denoising algorithm by selecting a large threshold λ, it inevitably decreases the second moment of the image reducing the brightness and contrast of the output. Unlike white Gaussian noise, DCT-domain quantization noise could contribute negatively to the second moment of the image, especially when the quality factor is low, hence, image denoising using NNM may pull the result further away from the statistics of the original image.

An intuitive solution to this problem is to completely preserve all the singular values that are above the threshold λ, i.e., to replace the soft-thresholding operator $\mathcal{D}_\lambda(\cdot)$ with a hard-thresholding operator

$$\mathcal{T}_\lambda(\boldsymbol{\Sigma})_{jj} = \begin{cases} \boldsymbol{\Sigma}_{jj} & \boldsymbol{\Sigma}_{jj} > \lambda, \\ 0 & \text{otherwise}. \end{cases} \tag{38}$$

Since $\mathcal{T}_\lambda(\boldsymbol{\Sigma})_{jj} > 0$ if and only if $\mathcal{T}_\lambda(\boldsymbol{\Sigma})_{jj} > 0$, the resulting matrices $\hat{\boldsymbol{X}}_i$ by the two threshold operators have the exact same rank. Thus, the new operator $\mathcal{T}_\lambda(\cdot)$ does not change the low rank property of the solution, however, in this case, the solution is closer to $\boldsymbol{Y}_i$ statistically in terms of the second moment.

This method coincides with the idea of reweighted nuclear norm minimization where large singular values are given smaller weight to achieve better low rank approximation [10]. It can also be interpreted as a spacial case of weighted nuclear norm minimization (WNNM) [8] as follows. If for $\sigma_j(\boldsymbol{X}_i)$, the j-th singular value of $\boldsymbol{X}_i$, we assign a weight w_j,

$$w_j = \begin{cases} 0 & \sigma_j(\boldsymbol{Y}_i) > \lambda, \\ \lambda & \text{otherwise}. \end{cases} \tag{39}$$

17

Since the weights $w_{1...M}$ are in a non-descending order, by the theory of WNNM, applying the hard-thresholding operator $\mathcal{T}_\lambda(\cdot)$ on $\mathbf{\Sigma}$ yields an optimal solution for optimization problem,

$$\hat{\mathbf{X}}_i = \underset{\mathbf{X}_i}{\text{argmin}} \|\mathbf{Y}_i - \mathbf{X}_i\|_F^2 + \|\mathbf{X}_i\|_{\boldsymbol{w},*} \tag{40}$$

where $\|\mathbf{X}_i\|_{\boldsymbol{w},*}$ is the weighted sum of the singular values of matrix $\mathbf{X}_i$.

Now, the question is how to set the parameter λ of the NNM problem making it more effective against DCT-domain quantization noise. In the formulation of Eq. (23), λ is a weight balancing the sparse and fidelity regularization terms. If sparsity is given too much weight, it tends to over-smooth the image and cause degradation in brightness and contrast as discussed previously; if the weight is too small, noise remains visible. From the perspective of the solution to the problem in Eq. (35), λ is a threshold eliminating small singular values of matrix $\mathbf{Y}_i = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\mathsf{T}}$, where row vector $\sigma_j \boldsymbol{v}_j^{\mathsf{T}}$ in matrix $\mathbf{\Sigma}\mathbf{V}^{\mathsf{T}}$ consists of the coefficient of each patch in $\mathbf{Y}_i$ with respect to the j-th basis vector in sparse dictionary $\mathbf{U}$ [7]. Image denoising by sparse optimization is based on the fact that signal is likely sparse under some basis while noise is i.i.d. under the same basis. Furthermore, the energy of compression error generally is small in comparison with the strength of signal, especially when the quantization factor is set to a practical range. Thus, removing small coefficients, which originated most likely from noise than signal, results an output closer to the true signal. The mean square error (MSE) ε^2 of compression can then be approximated by,

$$
\begin{aligned}
\varepsilon^2 &= \frac{1}{mM} \|\mathbf{Y}_i - \mathbf{X}_i\|_F^2 \\
&\approx \frac{1}{mM} \|\mathbf{Y}_i - \hat{\mathbf{X}}_i\|_F^2 \\
&= \frac{1}{mM} \|\mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\mathsf{T}} - \mathbf{U}\mathcal{T}_\lambda(\mathbf{\Sigma})\mathbf{V}^{\mathsf{T}}\|_F^2 \\
&= \frac{1}{mM} \|\mathbf{\Sigma} - \mathcal{T}_\lambda(\mathbf{\Sigma})\|_F^2 \\
&= \frac{1}{mM} \sum_{j \in L} \sigma_j^2
\end{aligned}
\tag{41}
$$

where set L contains indices of singular values that are less than λ. For the same input image, using a lower quality setting increases the compression noise (i.e., ε^2), which in turn increases the small singular values according to Eq. (41). To compensate this, threshold λ must increase as well to keep the size of set L unchanged so that the sparsity of the signal is preserved. Considering that the compression error is i.i.d. on each basis vector, implying that small singular values are of similar strength, threshold λ should be proportional to root mean square error (RMSE) ε as,

$$\varepsilon^2 \propto \frac{1}{mM} \sum_{j \in L} \lambda^2 \tag{42}$$

18

Therefore, with an empirical constant C_λ, threshold λ can be set as

$$\lambda = C_\lambda \varepsilon \sqrt{\frac{mM}{|L|}} \approx C_\lambda \varepsilon \sqrt{\max(M, m)}, \tag{43}$$

where we assume $|L| \approx \min(M, m)$ due to the sparsity of the true signal. This threshold selecting method requires the knowledge of the strength of the compression error ε, which is commonly unknown to the decoder. If error ε is indeed not provided by the encoder, various no-reference peak signal-to-noise ratio (PSNR) estimation techniques [19, 9, 1] can be used to estimate ε with sufficient accuracy for finding an appropriate threshold λ.

5.3   DCT Coefficient Constraint

Most iterative denoising techniques, such as [14, 7, 8], employ some regularization mechanisms to add a portion of filtered noise back to the denoised image in each iteration in order to reduce the loss of high frequency information as the result of multiple rounds of smoothing operators. The idea of adding noise back enables denoising techniques to remove large noise aggressively by oversmoothing the image in the first few iterations without completely removing the detail in the process. Then, during the following iterations, the image can be refined gradually using smoothing operators of lower strength. Mainly designed to deal with Gaussian noise, many of these above mentioned denoising techniques implement this iterative regularization by simply adding the difference between the observed noisy image and smoothed image back to the smoothed image, and use the result as the input noisy image for the next iteration.

For our DCT quantization noise reduction algorithm, this noise feedback method can be written as,

$$\begin{aligned}
\boldsymbol{y}_i^{(k)} &= \boldsymbol{z}_i^{(k)} + \delta(\boldsymbol{y}_i - \boldsymbol{z}_i^{(k)}) \\
&= \delta \boldsymbol{y}_i + (1 - \delta)\boldsymbol{z}_i^{(k)} \\
&= \boldsymbol{T}^{-1}[\delta \boldsymbol{T}\boldsymbol{y}_i + (1 - \delta)\boldsymbol{T}\boldsymbol{z}_i^{(k)}]
\end{aligned} \tag{44}$$

where coding block $\boldsymbol{y}_i$ is at location i in the observed image $\boldsymbol{y}$ as defined in Eq. (32), coding block $\boldsymbol{y}_i^{(k)}$ is the denoised version of $\boldsymbol{y}_i$ from the k-the iteration of the algorithm, coding block $\boldsymbol{z}_i^{(k)} = \tilde{\boldsymbol{R}}_i \boldsymbol{H} \boldsymbol{z}^{(k)}$ is the smoothed block at location i as in Eq. (30) and δ is a weight parameter adjusting the strength of noise feedback. As shown in Eq. 44, in DCT domain, this noise feedback process finds a weighted average between DCT coefficients $\boldsymbol{T}\boldsymbol{z}_i^{(k)}$ of the smoothed image and coefficients $\boldsymbol{T}\boldsymbol{y}_i$ of the noisy observation, adding image detail along with some reduced noise back to the result.

Similarly, the clipping operator $\mathcal{C}_\beta(\boldsymbol{\alpha}, \boldsymbol{\rho})$ introduced in the previous section in Eq. (33) has the effect of blending the smoothed image $\boldsymbol{\alpha}$ with DCT-domain quantization error tainted observation image $\boldsymbol{\rho}$ as well. By design, the clipping

operator finds an image that is close to image $\boldsymbol{\alpha}$ and has all of its DCT coefficients lying within the given quantization intervals $[\boldsymbol{QT\rho} - \beta, \boldsymbol{QT\rho} + \beta]$. The output image $\mathcal{C}_\beta(\boldsymbol{\alpha}, \boldsymbol{\rho})$ is closer to $\boldsymbol{\alpha}$ if threshold β is large, and it is closer to $\boldsymbol{\rho}$ if β is small. Therefore, with adjustable strength using parameter β, the clipping operator is also a suitable noise feedback function for our algorithm as follows,

$$\boldsymbol{y}_i^{(k)} = \mathcal{C}_\beta(\boldsymbol{z}_i^{(k)}, \boldsymbol{y}_i). \tag{45}$$

This formulation is the same as the solution to the DCT coefficient constraint problem in Eq. (32) except for the threshold β. Since applying the clipping operator multiple times is equivalent to applying it once with the smallest threshold β, i.e.,

$$\mathcal{C}_\beta(\mathcal{C}_{0.5}(\boldsymbol{z}_i^{(k)}, \boldsymbol{y}_i), \boldsymbol{y}_i) = \mathcal{C}_{\min(\beta, 0.5)}(\boldsymbol{z}_i^{(k)}, \boldsymbol{y}_i), \tag{46}$$

we only need to use the clipping operator once with threshold $\beta \leq 0.5$ to solve the DCT coefficient constraint problem and add filtered noise back to the result.

By the theory of narrow quantization constraint set (NQCS), the DCT coefficient clipping threshold β should be sufficiently small in order to achieve the optimal results in terms of PSNR [15]. For example, the authors of NQCS demonstrated that fixing threshold $\beta = 0.1$ is good enough for various images; several research papers on JPEG image deblocking and denoising reported that setting $\beta = 0.3$ often yields best results [23, 11, 17]. The best choice of β depends on the distributions of the DCT coefficients of the original image, quantization factors and characteristics of the smoothing technique. Although smaller threshold β generates PSNR-plausible results, it often brings blocking and ringing artifacts back to the result, deteriorating its perceptual visual quality.

To alleviate this problem, in the last iteration of our algorithm, instead of solving the optimization problem in Eq. (30), whose solution is given in Eq. (32) using the clipping operator, we solve a modified problem as follows,

$$
\begin{aligned}
\boldsymbol{y}^{(k)} = \underset{\boldsymbol{y}}{\arg\min} \quad & \|\nabla^2(\boldsymbol{y} - \boldsymbol{H}\boldsymbol{z}^{(k)})\|_2^2 \\
& + \alpha \sum_{i=1}^{n} \|\tilde{\boldsymbol{R}}_i(\boldsymbol{y} - \boldsymbol{H}\boldsymbol{z}^{(k)})\|_2^2 \\
\text{s.t.} \quad & |\boldsymbol{QT}\tilde{\boldsymbol{R}}_i\boldsymbol{y} - \boldsymbol{\gamma}_i| \leq \beta, \\
& i = 1 \dots n.
\end{aligned}
\tag{47}
$$

In addition to minimizing the difference between the smoothed image and output image, the objective function of this modified problem also minimizes the difference between their second order derivatives. This new regularization term encourages adding filtered noise back to locations that are discontinuous in the smoothed image, so that, artifacts are less noticeable in the output image perceptually. The modified problem in Eq. (47) is solvable using augmented Lagrangian method, which is more expensive than the clipping operator in Eq. (32) in terms of computational complexity. However, since our algorithm only solves this problem once during the last iteration, this technique can improve the visual quality of the output image without significantly increasing the overall cost.

20

Figure 10: Several widely used test images.

Alternatively, we can obtain the goal of eliminating the visual artifacts by adjusting DCT clipping threshold β and singular value threshold λ in the last two iterations. The idea is that, if in the last iteration K, most of the DCT coefficients of smoothed block $\boldsymbol{z}_i^{(K)}$ are already within the quantization intervals $[\boldsymbol{QTy}_i - 0.5, \boldsymbol{QTy}_i + 0.5]$, then artifacts cannot be reintroduced to the results by the clipping operator in Eq. (32) with threshold $\beta^{(K)} = 0.5$. To insure the condition that most DCT coefficients satisfy the quantization interval constraints, the strength of the smoothing operator must be reduced in the last iteration by using a smaller singular value threshold $\lambda^{(K)}$. By Eqs. (41) and (43), the standard deviation of the difference between the noisy input image $\boldsymbol{y}$ and smoothed image $\boldsymbol{z}^{(K)}$ is,

$$\frac{\|\boldsymbol{y} - \boldsymbol{z}^{(K)}\|_2}{\sqrt{n}} \approx \frac{\lambda^{(K)}}{C_\lambda \sqrt{\max(M, m)}}, \tag{48}$$

which is roughly proportional to threshold $\lambda^{(K)}$, thus, decreasing threshold $\lambda^{(K)}$ also reduces the variance of $\boldsymbol{y} - \boldsymbol{z}^{(K)}$ in DCT domain and makes DCT coefficients of image $\boldsymbol{z}^{(K)}$ more likely stay within quantization interval. On the other hand, the clipping threshold $\beta^{(K-1)}$ in the second last iteration should also be small in order to make each DCT coefficient of the clipped image close to the centre of quantization interval, limiting DCT coefficient overflow caused by the next smoothing operator. However, if both thresholds β and λ are too small, it weakens the effect of noise reduction. In practice, we find that setting clipping threshold $\beta^{(K-1)} = 0.2$ and singular value threshold $\lambda^{(K)} = \lambda^{(1)}/4$ works well for most input images.

6 Experimental Results

To demonstrate the performance of the proposed technique, we first turn off the image restoration part by setting degradation matrix $\boldsymbol{H}$ as an identity matrix,

Image	JPEG	ACR	BM3D	WNNM	TV	DicTV	DTPD	Proposed
Lenna	30.64	+0.23	+0.92	+0.66	+0.12	+0.48	+1.45	**+1.58**
Parrot	32.37	+0.28	+0.75	−2.85	−0.17	+0.41	+1.36	**+1.68**
Hat	31.47	+0.22	+0.89	−2.56	+0.21	+0.45	+1.39	**+1.53**
Flower	30.10	+0.16	+0.99	−0.48	+0.12	+0.32	+1.49	**+1.78**
Monarch	28.32	+0.07	+1.32	+1.94	+1.56	+1.39	+2.64	**+2.90**
Leaves	28.90	+0.26	+1.70	+1.17	+0.70	+1.69	+3.14	**+3.44**
Barbara	30.41	+0.15	+1.36	+1.43	−1.19	+1.20	+2.81	**+3.25**
Boat	31.65	+0.34	+1.17	−0.94	−0.27	+0.60	+1.55	**+1.84**
House	33.72	+0.39	+0.95	−13.98	+0.10	+0.14	**+1.70**	+1.60
Bike	27.22	+0.08	+1.03	+1.21	+0.06	+0.87	+1.98	**+2.27**
Median	30.53	+0.22	+1.01	+0.09	+0.11	+0.54	+1.62	**+1.81**

Table 1: PSNR gains (dB) of different denoising algorithms at QF = 25.

Image	JPEG	ACR	BM3D	WNNM	TV	DicTV	DTPD	Proposed
Lenna	32.96	+0.05	+0.80	+0.85	−0.05	+0.06	+1.44	**+1.70**
Parrot	34.80	+0.10	+0.64	−0.49	−0.39	−0.11	+1.39	**+1.63**
Hat	33.64	+0.05	+0.89	+0.01	+0.01	−0.01	+1.60	**+1.83**
Flower	32.43	+0.02	+0.98	+0.46	+0.12	−0.20	+1.68	**+2.08**
Monarch	30.73	+0.00	+1.32	+1.99	+1.75	+1.33	+2.62	**+3.12**
Leaves	31.64	+0.04	+1.78	+0.45	+0.64	+1.58	+3.29	**+3.86**
Barbara	33.56	+0.05	+1.28	+1.68	−1.75	+0.56	+2.51	**+2.98**
Boat	34.42	+0.09	+1.20	+0.89	−0.64	−0.11	+1.52	**+2.04**
House	35.79	+0.13	+0.81	−10.87	−0.11	−0.69	+1.48	**+1.52**
Bike	29.95	+0.01	+1.10	+1.45	−0.15	+0.84	+2.21	**+2.64**
Median	33.26	+0.05	+1.04	+0.66	−0.08	+0.03	+1.64	**+2.06**

Table 2: PSNR gains (dB) of different denoising algorithms at QF = 50.

and compare the results with the state-of-the-art denoising and JPEG arti-fact removal techniques. The comparison group is composed of the following methods: one JPEG deblocking method: the ACR algorithm [22]; two denois-ing methods: the BM3D algorithm [5] and WNNM algorithm [8]; and three JPEG soft-decoding methods: the TV algorithm [2], DicTV [4] algorithm and DTPD algorithm [12]. As the denoising approaches BM3D and WNNM are not designed specifically for dealing with JPEG compression noise, they cannot estimate the compression error from the input JPEG image but require an es-timation of the error variance as a user input. To make a fair comparison, we provide the true variance of the compression error to these methods as a known parameter, so their performances should reflect their best results in removing JPEG compression noise.

We select several widely used images in the literature as test images (thumb-nailed in Figure 10). All images are 256×256 in size. Tables 1, 2 and 3 list the PSNR results of the compared algorithms on the test images compressed using JPEG with QF set to 25, 50 and 80, respectively. As shown in the tables, the proposed technique improves over the hard-decoded JPEG by around 2dB in PSNR. It leads in PSNR gain in almost every test case and has more than 0.2dB advantage over the second best method. As a reference, we also list objective

Image	JPEG	ACR	BM3D	WNNM	TV	DicTV	DTPD	Proposed
Lenna	36.59	+0.00	+0.57	+0.53	−0.69	−1.25	+1.22	**+1.57**
Parrot	38.31	+0.01	+0.46	+0.49	−0.99	−1.45	+1.07	**+1.47**
Hat	37.27	+0.02	+0.77	+0.96	−0.71	−1.41	+1.65	**+2.14**
Flower	36.20	+0.00	+0.98	+1.29	−0.57	−1.56	+1.71	**+2.32**
Monarch	34.73	−0.00	+1.22	+1.87	+1.01	+0.11	+2.33	**+3.10**
Leaves	35.92	−0.01	+1.74	+2.63	−0.30	−0.02	+3.10	**+3.97**
Barbara	37.61	+0.00	+0.89	+1.20	−2.59	−1.56	+1.69	**+2.09**
Boat	38.38	+0.01	+1.04	+1.21	−1.55	−2.29	+1.04	**+1.77**
House	39.11	+0.01	+0.87	+0.90	−0.88	−2.52	+1.70	**+1.98**
Bike	34.53	+0.00	+1.12	+1.63	−1.19	−0.52	+2.03	**+2.70**
Median	36.93	+0.00	+0.94	+1.21	−0.80	−1.43	+1.69	**+2.12**

Table 3: PSNR gains (dB) of different denoising algorithms at QF = 80.

Image	JPEG	ACR	BM3D	WNNM	TV	DicTV	DTPD	Proposed
Lenna	0.8835	+0.0097	+0.0231	+0.0177	+0.0041	+0.0050	+0.0260	**+0.0310**
Parrot	0.9060	+0.0114	+0.0186	−0.0141	+0.0063	+0.0032	+0.0193	**+0.0243**
Hat	0.8752	+0.0094	+0.0221	+0.0021	+0.0058	+0.0058	+0.0272	**+0.0315**
Flower	0.8816	+0.0080	+0.0271	+0.0123	+0.0040	+0.0045	+0.0306	**+0.0382**
Monarch	0.8969	+0.0069	+0.0425	+0.0453	+0.0396	+0.0371	+0.0514	**+0.0536**
Leaves	0.9234	+0.0112	+0.0386	+0.0431	+0.0302	+0.0362	+0.0472	**+0.0487**
Barbara	0.9033	+0.0079	+0.0245	+0.0210	−0.0255	+0.0078	+0.0348	**+0.0395**
Boat	0.8905	+0.0102	+0.0254	+0.0188	−0.0004	+0.0045	+0.0293	**+0.0340**
House	0.8741	+0.0060	+0.0110	−0.0431	+0.0006	+0.0024	**+0.0168**	+0.0166
Bike	0.8798	+0.0055	+0.0259	+0.0180	+0.0019	+0.0134	+0.0401	**+0.0458**
Median	0.8870	+0.0087	+0.0250	+0.0179	+0.0040	+0.0054	+0.0299	**+0.0361**

Table 4: SSIM gains of different denoising algorithms at QF = 25.

Image	JPEG	ACR	BM3D	WNNM	TV	DicTV	DTPD	Proposed
Lenna	0.9221	+0.0021	+0.0107	+0.0071	−0.0011	−0.0094	+0.0144	**+0.0186**
Parrot	0.9374	+0.0033	+0.0084	−0.0031	−0.0017	−0.0108	+0.0096	**+0.0134**
Hat	0.9182	+0.0032	+0.0142	+0.0070	−0.0025	−0.0095	+0.0200	**+0.0236**
Flower	0.9253	+0.0014	+0.0174	+0.0107	+0.0029	−0.0076	+0.0218	**+0.0277**
Monarch	0.9300	+0.0009	+0.0282	+0.0294	+0.0278	+0.0229	+0.0350	**+0.0377**
Leaves	0.9533	+0.0013	+0.0248	+0.0240	+0.0203	+0.0211	+0.0299	**+0.0322**
Barbara	0.9456	+0.0019	+0.0130	+0.0132	−0.0174	−0.0056	+0.0181	**+0.0215**
Boat	0.9319	+0.0029	+0.0165	+0.0145	−0.0017	−0.0069	+0.0189	**+0.0237**
House	0.9103	+0.0015	+0.0040	−0.0369	−0.0085	−0.0166	+0.0100	**+0.0124**
Bike	0.9280	+0.0008	+0.0162	+0.0130	+0.0022	+0.0005	+0.0276	**+0.0321**
Median	0.9290	+0.0017	+0.0152	+0.0118	−0.0014	−0.0072	+0.0194	**+0.0237**

Table 5: SSIM gains of different denoising algorithms at QF = 50.

Image	JPEG	ACR	BM3D	WNNM	TV	DicTV	DTPD	Proposed
Lenna	0.9559	+0.0001	+0.0015	−0.0018	−0.0045	−0.0198	+0.0061	**+0.0079**
Parrot	0.9629	+0.0003	+0.0018	−0.0001	−0.0041	−0.0192	+0.0041	**+0.0061**
Hat	0.9580	+0.0010	+0.0053	+0.0031	−0.0060	−0.0214	+0.0109	**+0.0131**
Flower	0.9623	+0.0001	+0.0090	+0.0089	−0.0004	−0.0156	+0.0119	**+0.0150**
Monarch	0.9626	+0.0000	+0.0121	+0.0120	+0.0131	+0.0050	+0.0174	**+0.0191**
Leaves	0.9789	−0.0003	+0.0114	+0.0128	+0.0081	+0.0046	+0.0131	**+0.0147**
Barbara	0.9738	+0.0001	+0.0034	+0.0028	−0.0116	−0.0169	+0.0056	**+0.0073**
Boat	0.9659	+0.0002	+0.0066	+0.0065	−0.0053	−0.0204	+0.0069	**+0.0101**
House	0.9530	+0.0002	+0.0040	+0.0023	−0.0133	−0.0346	+0.0111	**+0.0121**
Bike	0.9679	+0.0000	+0.0087	+0.0086	−0.0011	−0.0118	+0.0130	**+0.0156**
Median	0.9627	+0.0001	+0.0060	+0.0048	−0.0043	−0.0180	+0.0110	**+0.0126**

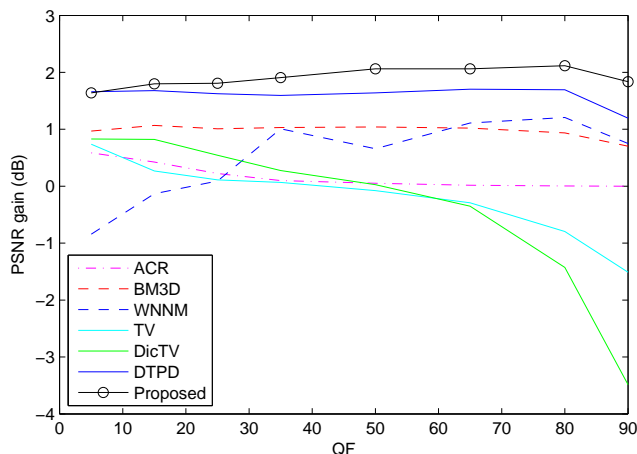Table 6: SSIM gains of different denoising algorithms at QF = 80.

Figure 11: The median PSNR gain as a function of QF.

fidelity assessment results by more sophisticated image quality metric SSIM [20] in Tables 4, 5 and 6 for different QF settings. As shown in the tables, the SSIM results also confirm the superiority of the proposed algorithm over the tested technologies.

Compared with other techniques, the proposed technique works consistently well at vastly different QF settings. As demonstrated in Figures. 11 and 12, the proposed technique is ahead of the competitions at all QF settings except when QF = 5. Only in that case, the proposed technique does not perform as well as DTPD in terms of median PSNR and SSIM gain. Although QF = 5 is often used in JPEG denoising research to showcase the capability of a technique, it has no practical value as compressing a down-scaled version of the input image with slightly larger QF could easily yield better output image than using QF = 5 directly. Furthermore, if we trade off time by increasing the number of iterations K, the proposed technique can outperform DTPD in both PSNR and
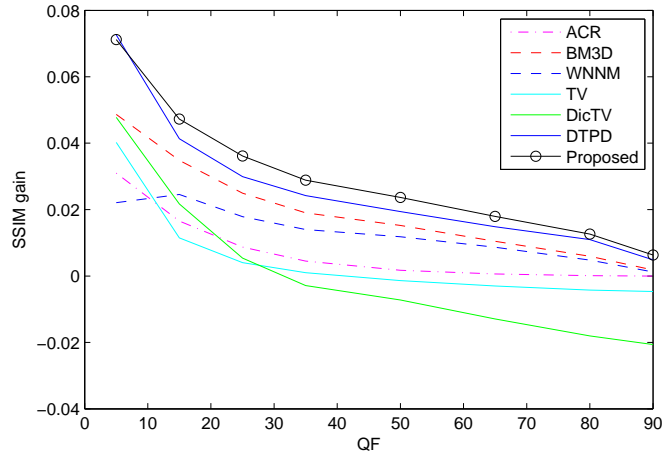
Figure 12: The median SSIM gain as a function of QF.

SSIM while still being faster than DTPD at QF = 5.

In addition to its superior performance in objective fidelity metric, the proposed approach also obtains better perceptual quality of the denoised images. As shown in Figures 13, 14 and 15 are some samples of the results from the tested algorithms. The output images of the proposed approach shows no discernible blocking and ringing artifacts even at low QF settings. The proposed approach preserves detail and edge structure visibly better than most of other techniques.

Like all the tested techniques except BM3D whose main functions are implemented in C++ and compiled to native code, the reference implementation of the proposed technique is written in pure MATLAB language, rendering it unfavourable in comparison of time cost with BM3D. Besides BM3D, the only other method faster than the proposed technique in the comparison group is ACR, which only reduces blocking artifacts and does not perform as well as most of the compared techniques in terms of either PSNR or SSIM.

7 Conclusion

Due to the low pass nature of image compression, the high-frequency components of a compressed image with sharp edges often carry large compression error. While high-frequency compression noise is relatively indiscernible in the original image as human visual system (HVS) is more sensitive to low-frequency noise, image restoration operator with high-boosting property can amplify the problem deteriorating the perceptive quality of restored image. By incorporating the non-linear DCT quantization mechanism into the formulation for image restoration, we propose new sparsity-based convex programming approach for

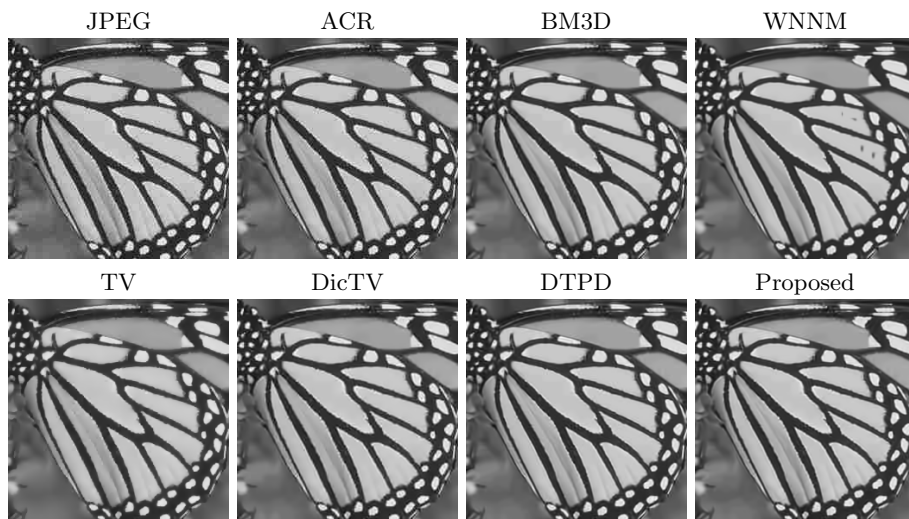Figure 13: Comparison of tested methods in visual quality at QF = 5.

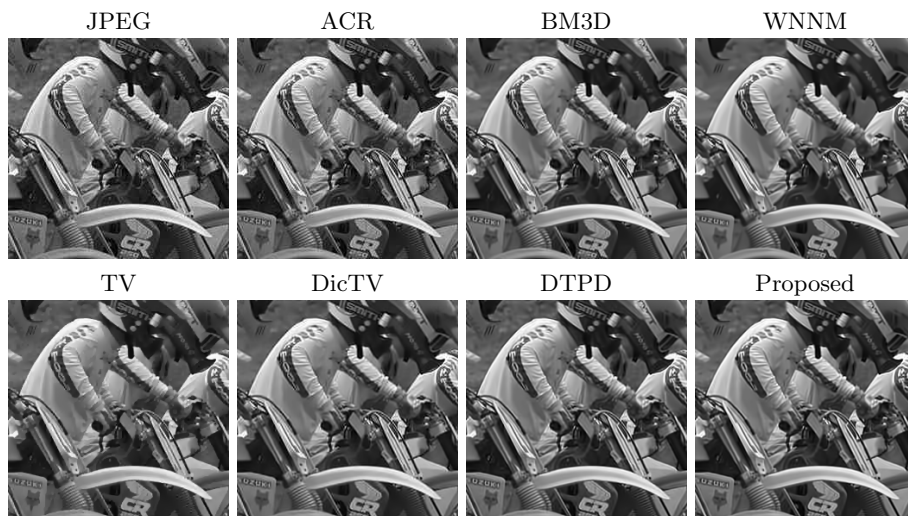Figure 14: Comparison of tested methods in visual quality at QF = 15.

Figure 15: Comparison of tested methods in visual quality at QF = 25.

joint quantization noise removal and restoration. Experimental results demonstrate significant performance gains of the new approach over existing restoration methods.

References

[1] T. Brandão and M. P. Queluz. No-reference image quality assessment based on DCT domain statistics. *Signal Processing*, 88(4):822–833, 2008.

[2] K. Bredies and M. Holler. A total variation-based JPEG decompression model. *SIAM Journal on Imaging Sciences*, 5(1):366–393, 2012.

[3] J.-F. Cai, E. J. Candès, and Z. Shen. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20(4):1956–1982, 2010.

[4] H. Chang, M. K. Ng, and T. Zeng. Reducing artifacts in JPEG decompression via a learned dictionary. *Signal Processing, IEEE Transactions on*, 62(3):718–728, 2014.

[5] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image denoising with block-matching and 3D filtering. In *Electronic Imaging 2006*, pages 606414–606414. International Society for Optics and Photonics, 2006.

[6] W. Dong, X. Li, L. Zhang, and G. Shi. Sparsity-based image denoising via dictionary learning and structural clustering. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 457–464. IEEE, 2011.

[7] W. Dong, G. Shi, and X. Li. Nonlocal image restoration with bilateral variance estimation: a low-rank approach. *Image Processing, IEEE Transactions on*, 22(2):700–711, 2013.

[8] S. Gu, L. Zhang, W. Zuo, and X. Feng. Weighted nuclear norm minimization with application to image denoising. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 2862–2869. IEEE, 2014.

[9] A. Ichigaya, M. Kurozumi, N. Hara, Y. Nishida, and E. Nakasu. A method of estimating coding PSNR using quantized DCT coefficients. *Circuits and Systems for Video Technology, IEEE Transactions on*, 16(2):251–259, 2006.

[10] Y.-F. Li, Y.-J. Zhang, and Z.-H. Huang. A reweighted nuclear norm minimization algorithm for low rank matrix recovery. *Journal of Computational and Applied Mathematics*, 263:338–350, 2014.

[11] A. W. Liew and H. Yan. Blocking artifacts suppression in block-coded images using overcomplete wavelet representation. *Circuits and Systems for Video Technology, IEEE Transactions on*, 14(4):450–461, 2004.

[12] X. Liu, X. Wu, J. Zhou, and D. Zhao. Data-driven sparsity-based restoration of JPEG-compressed images in dual transform-pixel domain. In *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*, pages 5171–5178, June 2015.

[13] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. Non-local sparse models for image restoration. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 2272–2279. IEEE, 2009.

[14] S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin. An iterative regularization method for total variation-based image restoration. *Multiscale Modeling & Simulation*, 4(2):460–489, 2005.

[15] S. H. Park and D. S. Kim. Theory of projection onto the narrow quantization constraint set and its application. *Image Processing, IEEE Transactions on*, 8(10):1361–1373, 1999.

[16] S. M. Pizer, E. P. Amburn, J. D. Austin, R. Cromartie, A. Geselowitz, T. Greer, B. ter Haar Romeny, J. B. Zimmerman, and K. Zuiderveld. Adaptive histogram equalization and its variations. *Computer vision, graphics, and image processing*, 39(3):355–368, 1987.

[17] D. Sun and W.-K. Cham. Postprocessing of low bit-rate block DCT coded images based on a fields of experts prior. *Image Processing, IEEE Transactions on*, 16(11):2743–2751, 2007.

[18] R. Timofte, V. De Smet, and L. Van Gool. A+: Adjusted anchored neighborhood regression for fast super-resolution. In *Computer Vision–ACCV 2014*, pages 111–126. Springer, 2014.

[19] D. S. Turaga, Y. Chen, and J. Caviedes. No reference PSNR estimation for compressed pictures. *Signal Processing: Image Communication*, 19(2):173–184, 2004.

[20] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *Image Processing, IEEE Transactions on*, 13(4):600–612, 2004.

[21] Q. Xie, D. Meng, S. Gu, L. Zhang, W. Zuo, X. Feng, and Z. Xu. On the optimal solution of weighted nuclear norm minimization. *arXiv preprint arXiv:1405.6012*, 2014.

[22] G. Zhai, W. Zhang, X. Yang, W. Lin, and Y. Xu. Efficient deblocking with coefficient regularization, shape-adaptive filtering, and quantization constraint. *Multimedia, IEEE Transactions on*, 10(5):735–745, 2008.

[23] G. Zhai, W. Zhang, X. Yang, W. Lin, and Y. Xu. Efficient image deblocking based on postfiltering in shifted windows. *Circuits and Systems for Video Technology, IEEE Transactions on*, 18(1):122–126, 2008.