A Contrast Enhancement Framework with JPEG Artifacts Suppression

Yu Li[1], Fangfang Guo[1], Robby T. Tan[2], and Michael S. Brown[1]

[1] National University of Singapore
[2] SIM University

Abstract. Contrast enhancement is used for many algorithms in computer vision. It is applied either explicitly, such as histogram equalization and tone-curve manipulation, or implicitly via methods that deal with degradation from physical phenomena such as haze, fog or underwater imaging. While contrast enhancement boosts the image appearance, it can unintentionally boost unsightly image artifacts, especially artifacts from JPEG compression. Most JPEG implementations optimize the compression in a scene-dependent manner such that low-contrast images exhibit few perceivable artifacts even for relatively high-compression factors. After contrast enhancement, however, these artifacts become significantly visible. Although there are numerous approaches targeting JPEG artifact reduction, these are generic in nature and are applied either as pre- or post-processing steps. When applied as pre-processing, existing methods tend to over smooth the image. When applied as post-processing, these are often ineffective at removing the boosted artifacts. To resolve this problem, we propose a framework that suppresses compression artifacts as an integral part of the contrast enhancement procedure. We show that this approach can produce compelling results superior to those obtained by existing JPEG artifacts removal methods for several types of contrast enhancement problems.

Keywords: Contrast Enhancement, Dehazing, JPEG Artifacts Removal, Deblocking

1 Introduction

A commonly applied procedure in low-level computer vision is contrast enhancement. This encompasses techniques that boost an image's global contrast through manipulations such as tone-curve adjustment, histogram equalization, and gradient-based enhancement. Such enhancement is beneficial for color segmentation, edge detection, image sharpening, image visualization, and many other tasks. In addition, spatially varying contrast enhancement is used to dramatically improve visibility in turbid media, such as haze, fog, rain, and underwater imaging.

Virtually all contrast enhancement algorithms operated on the assumption that the input image is uncompressed and free from significant noise. The reality, however, is that the vast majority of images available today on the internet
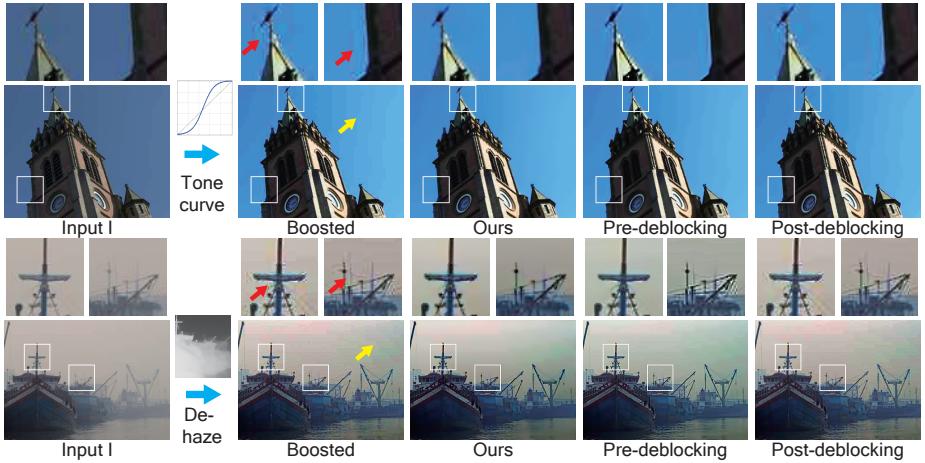
Fig. 1. This shows the noticeable compression artifacts after contrast enhancement. Top two rows are a tone-curve adjustment case ($Q40$) and the bottom two rows are a dehazing case ($Q70$). The zoomed-in regions are listed above to show the details. The characteristics of the blocking artifacts are distinctive in smooth regions (pointed out by the yellow arrows), while the ringing artifacts are along strong edges (pointed out by the red arrows). Comparison of our results with those of the deblocking method [10] applied before or after contrast enhancement results are shown. Note, our method produces more compelling results for reducing both blocking and ringing artifacts.

or from commodity imaging devices are compressed. Moreover, images coming from sources that would require contrast enhancement, e.g. surveillance cameras, often have notable amounts of image compression [13]. The most common compression scheme is by far JPEG and its extension to video, MPEG. The JPEG compression scheme breaks an input image into 8×8 pixel blocks and applies a discrete cosine transformation (DCT) to each block individually. To reduce storage space, the DCT coefficients are quantized at various levels – more quantization gives higher compression but lowers image quality (for more details see [25]). Lower-quality images exhibit what is termed collectively as "compression artifacts" that consist of the characteristic blocking artifacts resulting in discontinuities at the 8×8 borders, and oscillations or ringing artifacts next to strong edges.

Early JPEG compression methods use fixed quantization tables for different quality settings, however, most JPEG schemes now use what is referred to as *optimized JPEG* where quantization tables are customized based on the image's content [22]. This allows relatively high compression rates with little noticeable visual artifacts. However, when contrast boosting operations are applied, blocking and ringing artifacts become prominently visible as shown in Figure 1. [1]

---

[1] JPEG assigns a quality factor, QX, to indicate the subjective quality from 0 to 100 (from low quality to high quality)

There are several existing methods to reduce JPEG compression artifacts. These methods are often referred to as "deblocking" or "deringing". In the context of contrast enhancement, these methods would be applied either before or after the enhancement process. When applied before the enhancement process, the algorithms can smooth image details that have small contrast. When applied as a post-processing step, the effectiveness can be diminished due to the compression artifacts that were boosted by the contrast enhancement process. Figure 1 shows an example.

In this paper, we propose a framework based on structure-texture decomposition to remove the compression artifacts that are amplified in the image contrast enhancement operation. After the decomposition, contrast enhancement is directly applied to the structure layer, which is devoid of compression artifacts. Meanwhile, the texture layer, containing both image details and compress artifacts, is carefully processed to suppress only the artifacts. After proper scaling, the cleaned texture component is added back to the enhanced structure layer to generate the artifacts free output. Experimental results on various contrast enhancement task (*e.g.* Figure 1) demonstrate that our strategy can produce more compelling results (both qualitatively and quantitatively) than those of using general deblocking algorithms in either a pre- or post-processing manner. The details of our algorithm as well as comparisons with other methods are discussed in the following sections.

2 Related Work

We discuss relevant related work in the area of JPEG artifacts removal, contrast enhancement and multi-band image decomposition.

JPEG Artifacts Removal. JPEG artifacts, particularly blocking artifacts, have long been recognized in the image processing community (*e.g.* [15, 29]). Despite this, they remain unsolved and it is still an active area of research (*e.g.* [6, 28]). Various methods have been used, which can be broadly categorized into three different approaches. The first approach treats the compression artifacts as non-Gaussian noise and attempts to remove them by adaptive local filtering which adjusts the filter kernel to remove block edges and preserve image edges (*e.g.* [10]). The second approach is a reconstruction based approach that incorporates knowledge on natural images and encodes it into an energy function as a prior. Commonly used priors include spatial smoothness [27], quantization constraints, total variation (*e.g.* [11]), and gradient constraints (*e.g.* Field of Experts [19]). The third approach for reducing compression artifacts relies on machine learning techniques to learn a mapping from compressed images to their uncompressed version [14, 3]. While these approaches can reduce JPEG artifacts in images, their application as either a pre- or post-processing step can rarely outperform our method, which is designed explicitly for contrast enhancement.

Image Contrast Enhancement. Contrast enhancement can be performed in many ways. The most direct way is to apply a function f to the original pixel

intensity value, i.e. $I^e = f(I)$. This strategy is known as tone-curve adjustment. The function can be determined either manually or by selecting from pre-defined curves functions. Alternatively, the function can also be based on automatic histogram equalization, which obtains f by considering the input image's histogram. Aside from applying a certain function, local image gradients can also be used as a cost function that is optimized to boost contrast [17].

Recovering visibility in bad weather or underwater is, in fact, a specific contrast enhancement problem [20, 12, 5, 1]. Optically, poor visibility in bad weather or underwater is due to the substantial presence of medium particles that have significant size and distribution [20]. Light from the atmosphere and light reflected from an object are absorbed and scattered by those particles, leading to contrast reduction and thus to the degraded images. Most current dehazing algorithms try to estimate either airlight or transmission map (see [20, 9, 12]). Regardless the algorithms, the outputs of visibility enhancement show clear increase of contrast.

Multi-band Image Decomposition A common practice in solving computer vision and computational photography problems is to decompose images into different layers (or scales) and recombine them (*e.g.* multi-band image blending [4], optical flow estimation [26] *etc.*). The most related work to ours in this direction is tone-mapping methods (*e.g.* [7]), which attempt to reduce the contrast of a high dynamic range image to a limited range while preserving its details. This is usually achieved by reducing the contrast to the coarse layer and adding back the initial detail layer. Opposite to this tone-mapping methods' problem, however, we want to increase the contrast but not the noise/artifacts. As a result, we need to put more effort on processing the detail layer.

3   Proposed Method

Our basic pipeline is illustrated in Figure 2. It starts by decomposing the original input image into two layers: structure and texture layers. The input image can be considered as the superimposition of the two layers:

$$I = I_S + I_T, \qquad (1)$$

where I_S is the structure layer corresponding to the main large objects in the image, and I_T is the texture layer corresponding to the fine details [2]. The contrast of the structure layer is then enhanced according to our task (e.g., tone-curve adjustment or dehazing). The texture layer is processed through a combination of image matting and deblocking to remove compression artifacts. Finally the two layers are recombined to produce the final output. In the following, the details of each step are discussed.

3.1   Structure-Texture Decomposition

To decompose the input image into a structure layer and texture (high-frequency) layer, any edge-aware smoothing operation (e.g. bilateral filter [21], weight least
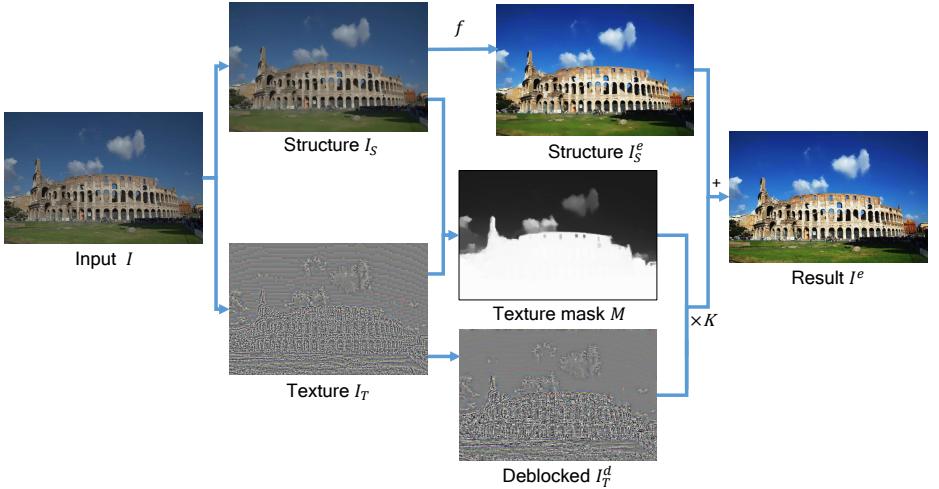
Fig. 2. The overview of our proposed method. The input image is decomposed into structure and texture components. The contrast of the structure component is then boosted directly; the texture component that contains the JPEG artifacts is processed to reduce compression artifact. The two components are recombined at the last step to render the final result. In the paper, we amplified the textural part by a factor of 10 and shift it by $+0.5$ for better visualization.

square filter [8]) can be applied. This procedure produces an image that retains strong structure and over-smooths out details. We take this image as the structure layer I_S, and obtain the texture layer by calculating the difference between the input image and its structure layer, $I_{T_i} = I_i - I_{S_i}$.

In our problem we applied the the total-variation (TV) image-reconstruction formulation based on Rudin-Osher-Fatemi method [18]. Based on the TV regularization, the structure layer I_S is obtained by minimizing the following objective function:

$$\min_{I_S} \sum_i (I_{S_i} - I_i)^2 + \lambda |\nabla I_{S_i}|, \tag{2}$$

where i is the pixel index, λ is the regulation parameter and ∇ is the gradient operator. An efficient half-quadratic splitting scheme to solve Eqn.(2) is described in [23].

This structure-texture decomposition exploits the fact that most of the structure layer is related to larger gradient magnitudes, while the texture layer captures both fine image details and compression artifacts that exhibit smaller gradient magnitudes. The parameter λ is important for controlling this separation and needs to be adjusted according to the compression factor, i.e., more compression requires λ to be increased. We show the values of λ used for different compression levels in the experiments section. There are methods for deblocking using TV regularization (*e.g.* [11]). The main difference here is that they do not explicitly process the texture layer, while our method put significant effort on
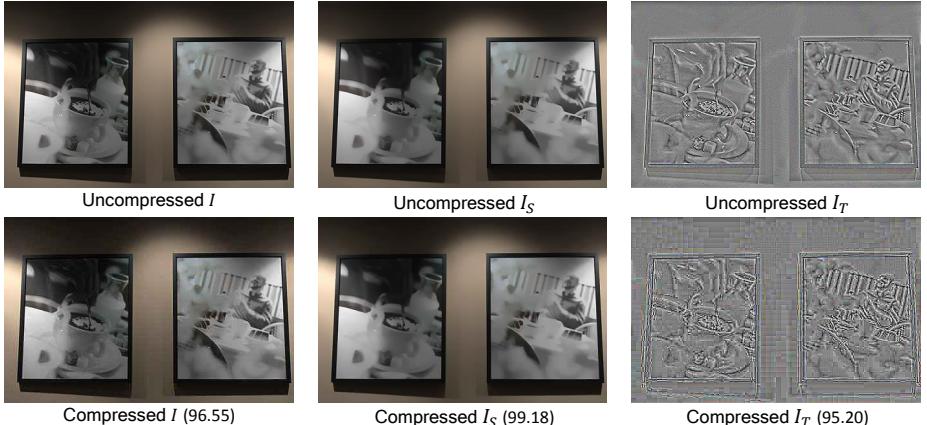
Fig. 3. This shows two examples of structure-texture decomposition in uncompressed and compressed image ($Q40$) pairs. The structure similarity index measurement (SSIM) [24] values (in $\times 100$ scale in the paper) between each pair are shown in the brackets. Notice that most of the characteristic compression artifacts exist in the texture layer, while the structure layer of the compressed image resembles that of the uncompressed image.

processing the texture layer as will be described later. As a result, TV-based deblocking methods tend to suffer from over-smoothing, while ours preserves more details.

Figure 3 shows two examples of the structure-texture decomposition results for the same images: one image is compressed and the other is not. As can be observed, unlike the texture layers that contain different information due to the artifacts, the structure layers are almost identical (both from the visual quality perspective and from the structure similarity index measurement, SSIM, perspective [24]). This shows the effectiveness of the TV regularization in producing a structure layer that significantly filter out any compression artifacts. As such, this image layer is considered to be artifacts free and suitable to be boosted using the desired enhancement operation directly, resulting in the enhanced version of the structure, I_S^e.

3.2   Reducing Artifacts in the Texture Layer

Since the texture layer contains both scene details and compression artifacts, it needs further refinement to be able to remove artifacts and to keep scene details. To do this, we create a mask M that separates regions, where the most scene details are presence, from the remaining regions. Having created the mask, for the regions inside the mask, we refine them further to remove potential ringing and blocking artifacts. For the remaining regions, which are those outside the mask, we remove the content altogether, since the content is most likely compression artifacts.
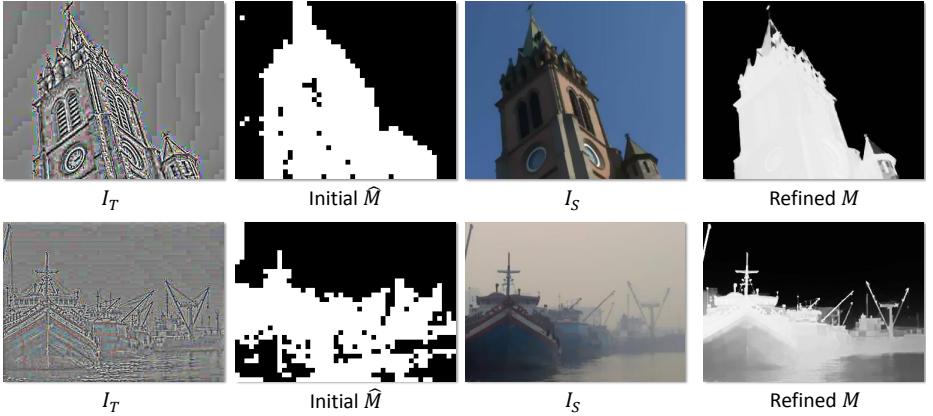
Fig. 4. This shows two examples of the scene detail map generation. The initial results obtained by checking DCT coefficients are rough estimations. A soft matting technique can help refine the map by applying it to the structure layer, and the result is well aligned with the objects in the images

Scene Detail Extraction To create the image mask, M, we apply the discrete cosine transform (DCT) to each 8×8 patches in the texture layer. We use the DCT high-frequency layer to serve as a likelihood of the scene details, i.e. stronger high-frequency DCT coefficients means more details. Denoting the 8×8 DCT of one block as matrix B, then the likelihood of this block to be part of the scene details can be expressed as:

$$t = \sum_{u,v} B_{u,v}^2 - B_{1,1}^2 - B_{1,2}^2 - B_{2,1}^2, \tag{3}$$

where u, v denotes the position in the DCT. We take the sum of squares of all D-CT coefficients except $B_{1,1}, B_{1,2}$ and $B_{2,1}$, and apply a threshold to the likelihood to make a binary indication of each block. The threshold we use is empirically set to 0.1. This initial block-wise estimation of texture region, denoted as $\hat{M}$, is a coarse estimate, as shown in the second column of Figure 4.

This initial mask provides the regions of image details, but is too coarse for practical use. Thus, we apply a refinement step to better align the texture region with the structure layer. For this, we use a soft matting technique (inspired by [12]) by minimizing the following function on the scene detail map M:

$$\min_{\mathbf{m}} (\mathbf{m} - \hat{\mathbf{m}})^\top (\mathbf{m} - \hat{\mathbf{m}}) + \alpha \mathbf{m}^\top L_s \mathbf{m}, \tag{4}$$

where $\mathbf{m}$ and $\hat{\mathbf{m}}$ are the vector forms of matrix M and $\hat{M}$, respectively. L_s is Levin's [16] matting Laplacian matrix generated from I_S. The smallest eigenvectors of the matting Laplacian correspond to the partitioning of images [16]. The first term forces the agreement with the initial estimation $\hat{M}$, while the second term forces the output to be aligned with the structure layer I_S. We set the

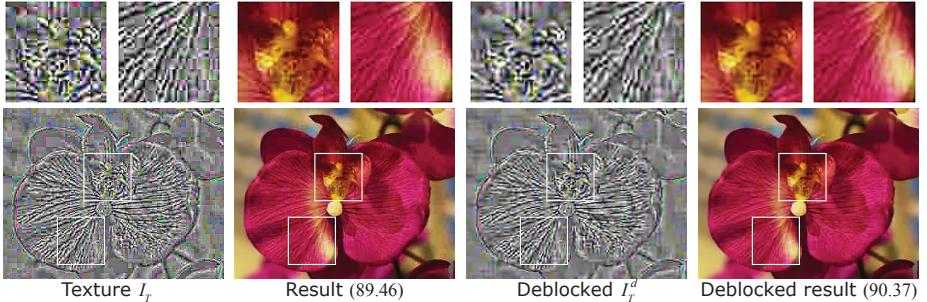| Texture I_T | Result (89.46) | Deblocked I_T^d | Deblocked result (90.37) |

Fig. 5. This shows the effect of blocking artifacts reduction. The left side shows the textural layers and its corresponding final composition results without the blocking artifacts reduction step. The right side shows the same pair but with the effect of blocking artifacts reduction. As can be seen both in texture and final results that the block is less noticeable when we apply the block artifacts reduction. The similarity against ground truth using SSIM for with and without deblocking are also shown in the bracket.

regularization parameter α a large value (10^5 in our implementation), since it will provide clearer edges in the mask M. The last column of Figure 4 shows the texture region map after refinement using the structure I_S. Most of the values in the map are near 0 or 1 (close to binary), but some values are between the two.

The result is a mask M whose edges have been refined. Another benefit of aligning the mask to the structure layer is that small amounts of textures around edges, which are indicative to ringing artifacts, are removed.

Block Artifacts Reduction Having created the mask indicating the regions of scene details, we now try to reduce the potential blocking artifacts in the regions. Denoting the texture image after blocking artifacts suppression as $I_{T_i}^d$, an objective function is defined as follows:

$$\min_{I_{T_i}^d} \sum_i (I_{T_i}^d - I_{T_i})^2 + \beta \sum_{i \in \eta} (\nabla I_{T_i}^d)^2, \tag{5}$$

where i is the pixel index, and η are the locations at the 8×8 block borders. The first term forces the output to be similar to the input, while the second term smooths the edges at the 8×8 block borders, since they are more likely to be block artifacts. The smoothness level is controlled by the weight term β. We empirically set it 0.5 to achieve a proper compromise between oversmoothness and noticeable artifacts. This is effective in reducing the blockings in the texture map and result in a higher quantitative score as can be seen in Figure 5.

3.3 Layer Recomposition

Having removed the artifacts in the texture layer, we now need to apply an enhancement operation to the texture layer before adding it back to I_S^e. However, since most contrast functions f are not linear and thus $f(I_S + I_T) \neq f(I_S) + f(I_T)$, we cannot simply apply the same process and then sum them up. As a consequence, we have to approximate the enhancement function adjustment by finding a scale multiplication factor K, which should obey the following condition as much as possible: $f(I) = f(I_S) + KI_T$, where I is the original input image. By denoting the enhanced texture layer as I_T^e, we intend to find the scale factor K:

$$I_T^e = K \circ M \circ I_T^d, \tag{6}$$

where $\circ$ is the element-wise multiplication operator. $M \circ I_T^d$ combines the steps in the previous section that generates the masked texture layer with reduced artifacts.

Like in the case of enhancing contrast for the structure layer I_S, the scale factor depends on the applications. For the application of image tone-curve adjustment, the tone-curve function f is applied to the intensity values of the input image, I. Taylor series $f(t + \Delta t) \approx f(t) + f'(t)\Delta t$ allows us to write:

$$f(I_{S_i} + I_{T_i}) = f(I_{S_i}) + f'(I_{S_i})I_{T_i}. \tag{7}$$

Hence, from the last equation, we have the scale factor for the tone adjustment $K_i = f'(I_{S_i})$.

In the dehazing or underwater application, the enhancement should consider the optical model of scattering media, which according to [12], the output of the enhancement should follow the following equation:

$$I_i^e = \frac{I_i - A}{t_i} + A, \tag{8}$$

where the I_i is the input image, A is the atmospheric light, t_i is the transmission, and i is pixel index. Therefore, the scale factor, K_i, should be approximately equal to $\frac{1}{t_i}$, since A is a constant and I_i^e is in $\frac{I_i}{t_i} + k$ form. Following [12], t is obtained from dark channel prior and A is obtained from the patch with the brightest intensity in dark channel.

Having recovered both the structure and texture layers, the final result can be achieved by simply summing up the two layers: $I^e = I_S^e + I_T^e$.

4 Results

We evaluated our proposed framework by applying it to various contrast enhancement tasks: image tone-curve adjustment, dehazing and underwater visibility enhancement. Due to space limitation, only some of our results are shown here. More results are available in the supplemental material. Demo code and

Low contrast ($Q40$)   Haze ($Q80$)   Underwater ($Q30$)

Fig. 6. This figure shows the inputs in the section which require contrast boosting.

Table 1. Average Runtime Comparison

Method	SA-DCT[10]	FoE[19]	NN[3]	Ours
Runtime(s)	20	287	25	15

data are available at the author's webpage [2]. Experiments were performed on a PC with Intel I7 CPU (3.4GHz) with 8GB RAM. The test images were either self-taken or downloaded from the Internet. Three examples are shown in Figure 6. Note that, in these input images, there are often no noticeable artifacts. The artifacts become apparent after the contrast enhancement is applied.

The entire process for an image (approximately 500×600 in size) using our current un-optimized matlab implementation took about 15 seconds with the main bottleneck being the image matting which took more than 10 seconds. The only parameter that needs to be changed was the regulation term λ in the structure-texture decomposition in Eqn. (2). This parameter was set according to the compression level. Higher compression requires larger λ for the decomposition. $\lambda = 0.02, 0.03, 0.04, 0.05$ is used for $> Q70$, $Q50 - Q70$, $Q30 - Q50$ and $< Q30$, respectively.

We compared our approach with several state-of-the-art deblocking methods: a local filtering based method - shape adaptive DCT (SA-DCT) [10], a reconstruction based using Field of Experts (FoE) prior [19] as well as a learning based method using Neural Network (NN) [3]. These methods were all used as both a pre-processing and post-processing step for the contrast enhancement methods. We note that the comparison with NN is not fair since it is a more general algorithm targeting on any kinds of noise (i.e. not just JPEG artifacts). The average run-times of these algorithms are summarized in Table 1. Interest-

---

[2] http://www.comp.nus.edu.sg/ liyu1988/

Table 2. Quantitative Comparison

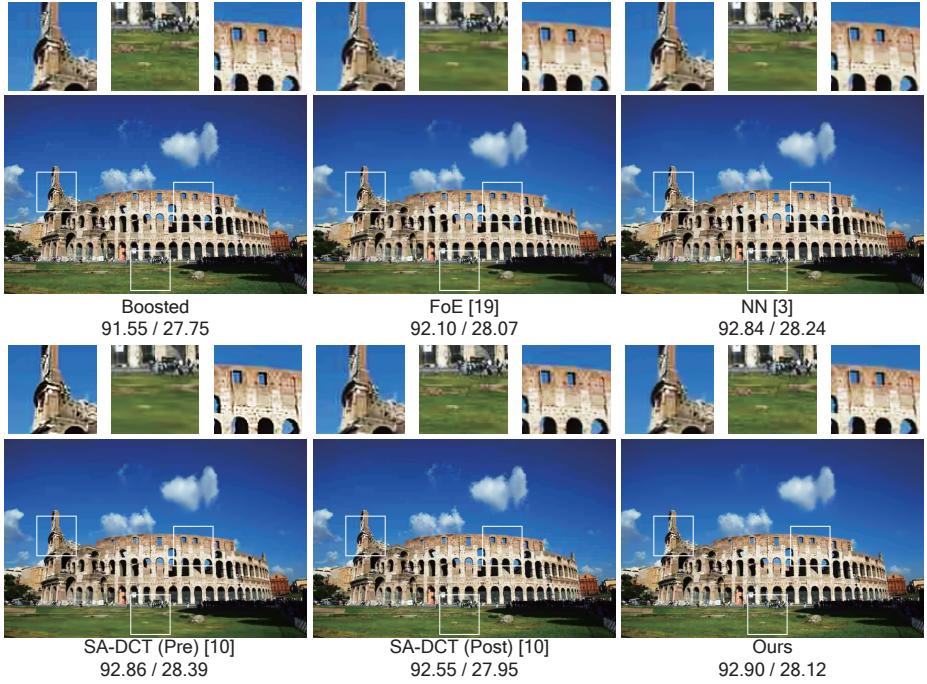| Method	simple boosted	FoE[19]	NN[3]	SA-DCT (Pre)[10]	SA-DCT (Post)[10]	Ours
Mean SSIM	90.79	91.14	91.88	92.03	91.79	**92.05**
Mean PSNR	29.17	29.69	29.94	**30.12**	29.42	29.76

Fig. 7. This figure shows an example in image tone-curve adjustment using FoE [19], NN [3], SA-DCT [10] and our approach. Shown below the images are the comparison SSIM/PSNR(dB) with respect to the groundtruth.

ingly, even though we apply layer decomposition and matting as parts of our procedure, our method has the fastest performance among all.

For experiments involving tone-curve manipulation, we can also provide a quantitative comparison with the groundtruth. The groundtruth image is obtained by enhancing the uncompressed image using the same tone-curve. Quantitative results are reported using the perceptually-based quality measurement-structure similarity index (SSIM) [24] (in $\times 100$ scale) as well as the peak signal-to-noise ratio (PSNR). Table 2 summarizes the average SSIM and average PSNR on all our 15 test cases and at different compression levels (from $Q20$ to $Q90$). Our approach achieves the highest SSIM but not the highest PSNR. As sometimes the case with PSNR, we believe it does not properly reflect the qualitative results. On visual inspection of the images, it is clear our approach is qualitatively better than the other methods.

Figure 7 shows a tone-curve adjustment comparison. As can be seen, FoE and NN successfully removed block artifacts which resulted in overall improvements in both PSNR and SSIM. However, they tended to smooth sharp edges and details in the image. SA-DCT lost its effectiveness in deblocking when used after the enhancement, but when used before the enhancement, SA-DCT did a good
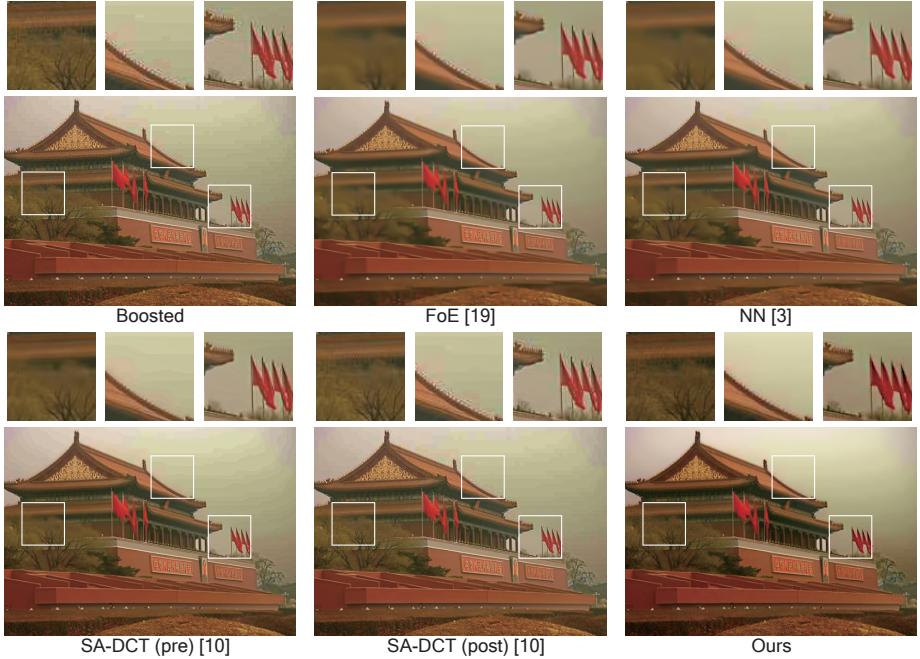
Fig. 8. This figure shows an example of dehazing using FoE [19], NN [3], SA-DCT [10] and our approach.

job and achieved the highest PSNR. However, upon close visual inspection, the results of our method are much cleaner (less ringing artifacts) and more image details preserved, resulting the highest SSIM value.

Figures 8 and 9 show examples of applying our method to dehazing and underwater visibility enhancement. Here, since we do not have the groundtruth recovered image, we can only show qualitative visual comparisons. In these applications, the advantage of our method becomes more observable. The results of using FoE and NN are over smoothed, causing them to lose details. SA-DCT slightly outperformed FoE and NN in reducing the compression artifacts. Ours is better in terms of removing artifacts (particularly with much less ringings) as well as preserving image details.

Due to the nature of this problem, the results are best viewed in the original size. Thus, we provide larger images as well as more comparisons and results in the supplemental material.

5   Discussion and Conclusion

We have introduced a framework to suppress artifacts appearing in JPEG images that becomes prominently visible when applying contrast enhancement. While the proposed framework is admittedly engineering in nature, our strategy of
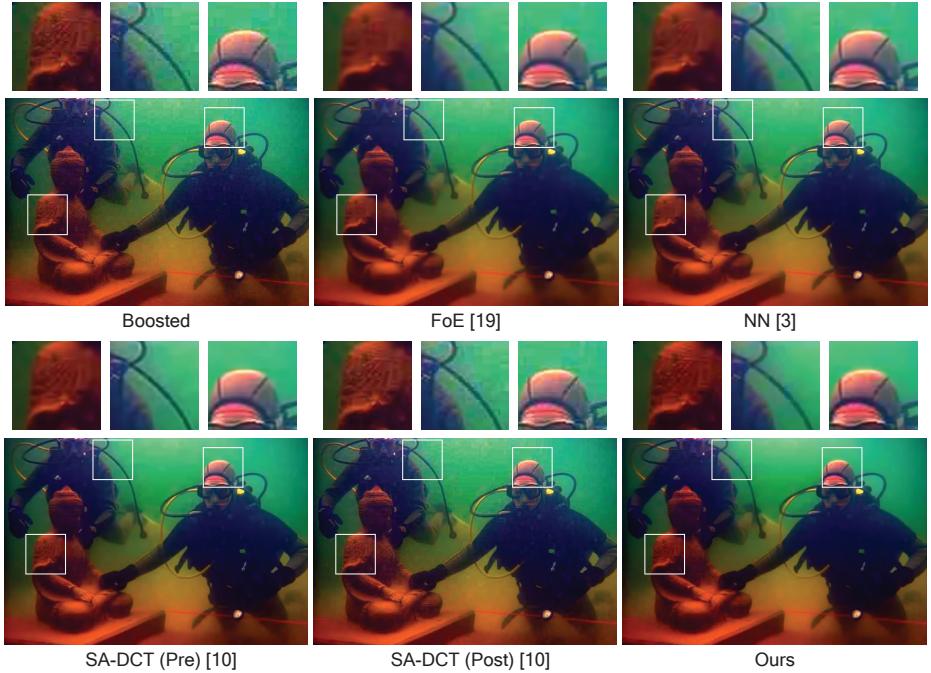
Fig. 9. This figure shows an example of underwater image enhancement using FoE [19], NN [3], SA-DCT [10] and our approach.

using structure and texture layer decomposition enables us to reduce the compression artifacts in parallel with contrast enhancement, and to process them independently to each other. With this integrated framework, the key benefit is that we can process two tasks that are opposite to each other in terms of functionality. On one hand, we have a task to suppress noise as much as possible; on the other hand, within the same image, we have a task to enhance the content as much as possible. If these two tasks are processed sequentially, as pre- or post-processing, the results are not likely to be optimum. Since, the process of artifacts removal as pre-processing will remove the image content that have low contrast, and as post-processing will be affected by the enhanced artifacts. As shown in our experiments, we have demonstrated the effectiveness of the proposed framework using qualitative and quantitative measures.

While our approach targets suppressing JPEG compression artifacts for the task of contrast enhancement, our framework is suitable to other applications that have the same nature of problem. We consider JPEG compression artifacts to be an important problem because these are commonly troublesome for many computer vision and image processing algorithms that assume the input images have little noise. We also consider contrast enhancement, since it is one of the core operations in the low-level computer vision and image processing. Among

other applications, it is crucially used to deal with turbid media, such as haze, fog, rain, and underwater, which has been addressed considerably in computer vision community recently.

Regarding our framework, the remaining question is whether our structure, texture, and masked texture layers can effectively distill JPEG images into a layer that is mostly image content and also into another layer that is mostly affected by compression artifacts. While our practical findings discussed in the paper have given us a positive answer (and we consider as a contribution that can be improved further), rigorous evaluation is still needed, and we will consider this in our future work.

6 Acknowledgement

References

1. Ancuti, C., Ancuti, C.O., Haber, T., Bekaert, P.: Enhancing underwater images and videos by fusion. In: IEEE Conference on Computer Vision and Pattern Recognition (2012)
2. Aujol, J.F., Gilboa, G., Chan, T., Osher, S.: Structure-texture image decompositionmodeling, algorithms, and parameter selection. International Journal of Computer Vision 67(1), 111–136 (2006)
3. Burger, H.C., Schuler, C.J., Harmeling, S.: Image denoising: Can plain neural networks compete with bm3d? In: IEEE Conference on Computer Vision and Pattern Recognition (2012)
4. Burt, P.J., Adelson, E.H.: The laplacian pyramid as a compact image code. IEEE Transactions on Communications 31(4), 532–540 (1983)
5. Chiang, J.Y., Chen, Y.C.: Underwater image enhancement by wavelength compensation and dehazing. IEEE Transactions on Image Processing 21(4), 1756–1769 (2012)
6. Dong, W., Zhang, L., Shi, G.: Centralized sparse representation for image restoration. In: IEEE International Conference on Computer Vision (2011)
7. Durand, F., Dorsey, J.: Fast bilateral filtering for the display of high-dynamic-range images. ACM Transactions on Graphics (TOG) 21(3), 257–266 (2002)
8. Farbman, Z., Fattal, R., Lischinski, D., Szeliski, R.: Edge-preserving decompositions for multi-scale tone and detail manipulation. ACM Transactions on Graphics (TOG) 27(3), 67 (2008)
9. Fattal, R.: Single image dehazing. ACM Transactions on Graphics 27(3), 72 (2008)
10. Foi, A., Katkovnik, V., Egiazarian, K.: Pointwise shape-adaptive dct for high-quality denoising and deblocking of grayscale and color images. IEEE Transactions on Image Processing 16(5), 1395–1411 (2007)
11. Goto, T., Kato, Y., Hirano, S., Sakurai, M., Nguyen, T.Q.: Compression artifact reduction based on total variation regularization method for mpeg-2. IEEE Transactions on Consumer Electronics 57(1), 253–259 (2011)

12. He, K., Sun, J., Tang, X.: Single image haze removal using dark channel prior. IEEE Transactions on Pattern Analysis and Machine Intelligence 33(12), 2341–2353 (2011)
13. Jacobs, N., Burgin, W., Fridrich, N., Abrams, A., Miskell, K., Braswell, B.H., Richardson, A.D., Pless, R.: The global network of outdoor webcams: Properties and applications. In: ACM International Conference on Advances in Geographic Information Systems (2009)
14. Lee, K., Kim, D.S., Kim, T.: Regression-based prediction for blocking artifact reduction in jpeg-compressed images. IEEE Transactions on Image Processing 14(1), 36–48 (2005)
15. Lee, Y., Kim, H., Park, H.: Blocking effect reduction of jpeg images by signal adaptive filtering. IEEE Transactions on Image Processing 7(2), 229–234 (1998)
16. Levin, A., Lischinski, D., Weiss, Y.: A closed-form solution to natural image matting. IEEE Transactions on Pattern Analysis and Machine Intelligence 30(2), 228–242 (2008)
17. Majumder, A., Irani, S.: Perception-based contrast enhancement of images. ACM Transactions on Applied Perception 4(3), 17 (2007)
18. Rudin, L.I., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. Physica D: Nonlinear Phenomena 60(1), 259–268 (1992)
19. Sun, D., Cham, W.K.: Postprocessing of low bit-rate block dct coded images based on a fields of experts prior. IEEE Transactions on Image Processing 16(11), 2743–2751 (2007)
20. Tan, R.T.: Visibility in bad weather from a single image. In: IEEE Conference on Computer Vision and Pattern Recognition (2008)
21. Tomasi, C., Manduchi, R.: Bilateral filtering for gray and color images. In: IEEE International Conference on Computer Vision. pp. 839–846 (1998)
22. Wang, C.Y., Lee, S. M. Chang, L.W.: Designing jpeg quantization tables based on human visual system. Image Communication 16(5), 501–506 (2001)
23. Wang, Y., Yang, J., Yin, W., Zhang, Y.: A new alternating minimization algorithm for total variation image reconstruction. SIAM Journal on Imaging Sciences 1(3), 248–272 (2008)
24. Wang, Z., Bovik, A., Sheikh, H., Simoncelli, E.: Image quality assessment: From error visibility to structural similarity. IEEE Transactions on Image Processing 13(4), 600–612 (2004)
25. Watson, A.: Dct quantization matrices visually optimized for individual images. In: Proceedings of the International Society for Optics and Photonics. vol. 1913, pp. 202–216 (1993)
26. Wedel, A., Pock, T., Zach, C., Bischof, H., Cremers, D.: An improved algorithm for tv-l 1 optical flow. Statistical and Geometrical Approaches to Visual Motion Analysis pp. 23–45 (2009)
27. Yang, Y., Galatsanos, N.P., Katsaggelos, A.K.: Projection-based spatially adaptive reconstruction of block-transform compressed images. IEEE Transactions on Image Processing 4(7), 896–908 (1995)
28. Yim, C., Bovik, A.: Quality assessment of deblocked images. IEEE Transactions on Image Processing 20(1), 88–98 (2011)
29. Zakhor, A.: Iterative procedures for reduction of blocking effects in transform image coding. IEEE Transactions on Circuits and Systems for Video Technology 2(1), 91–95 (1992)