Visual data deblocking using structural layer priors

Conference Paper · July 2016

DOI: 10.1109/ICME.2016.7552893

3 authors, including:

Jiawan Zhang
Tianjin University
101 PUBLICATIONS **404** CITATIONS

SEE PROFILE

VISUAL DATA DEBLOCKING USING STRUCTURAL LAYER PRIORS

Siyuan Li[‡]*, Jiawan Zhang*[‡] *and Xiaojie Guo*[†*]

[‡]School of Computer Software, Tianjin University
[†]State Key Laboratory Of Information Security, IIE, CAS
siyuan@tju.edu.cn jwzhang@tju.edu.cn xj.max.guo@gmail.com

ABSTRACT

The blocking artifact frequently appears in compressed real-world images or video sequences, especially coded at low bit rates, which is visually annoying and likely hurts the performance of many computer vision algorithms. A compressed frame can be viewed as the superimposition of an intrinsic layer and an artifact one. Recovering the two layers from such frames seems to be a severely ill-posed problem since the number of unknowns to recover is twice as many as the given measurements. In this paper, we propose a simple and robust method to separate these two layers, which exploits structural layer priors including the gradient sparsity of the intrinsic layer, and the independence of the gradient fields of the two layers. A novel Augmented Lagrangian Multiplier based algorithm is designed to efficiently and effectively solve the recovery problem. Experimental results demonstrate the efficacy of our method.

Index Terms— Deblocking, Structural Layer Priors

1. INTRODUCTION

With the emergence of mobile devices, the amount of user captured and shared images and videos rapidly increases. A huge space for storing and a wide bandwidth for transmitting such data are required if without reducing their file sizes properly. Image and video compression techniques have been designed to reduce the file size meanwhile preserve the visual quality of the frames. JPEG [1], MPEG and H.26x [2] are classic and widely used standards in its history, which employ the block Discrete Cosine Transform (DCT), due to its good energy compaction and decorrelation properties, to achieve the compression. However, an inevitable problem of these standards is that as the compression ratio increases, the fidelity of coded images degrades, *i.e.* details are ruined and artificial block boundaries appear (see Fig. 1 for example).

Considering the flexibility to existing codecs makes post-processing approaches attractive, which handle compressed frames at the decoder end, without changing the maturing structure of existing codecs. In the last decades, significant research has been made towards the development of post-processing style deblocking techniques, which can be broadly categorized into two different groups, namely the heuristic deblocker and the optimization-based one.

The heuristic deblockers attempt to suppress the artifact by (adaptive) local filters. Very first work proposed by Lim and Reeve [4] employs the low pass filter on boundaries, which may also blur intrinsic edges of the image. To address this problem, techniques that adaptively perform filtering on regions obtained by either classification or detection have been proposed [5]. The recent video coding standard, H.264/AVC [2], analyzes artifacts and chooses different filters for different block boundaries according to their local properties. These filtering methods consider the artifacts as noises to be smoothed for visual improvement. However, in general, this kind of deblockers aims at heuristically smoothing visible artifacts without objective criterion, instead of genuinely restoring the original information.

Alternatively, the optimization-based methods focus on recovering the intrinsic layer under some assumptions. Jung *et al.* attempt to reconstruct the intrinsic layer via sparse representation, which, however, requires the compression ratio is known and the dictionary is well-learned [6]. More recently, Sun and Liu [7] introduce a non-causal temporal prior for video deblocking, which iteratively refines the target frames and the estimation of motion across them. Due to the iterative procedure and optical flow estimation, its computational load is heavy. Li *et al.* [3] develop a four-step method including structure-texture decomposition, scene detail extraction, block artifact reduction and layer recomposition. This approach favors the whole or a big part of image with poor texture. Usually, the recovered results obtained by the restoration-style methods are of better quality than those by the denoising ones. But they are typically time consuming or complex.

As can be seen from the aforementioned methods, the characteristics of the two layers have been well investigated individually, the relationship between the two layers, however, has been rarely studied. This work exploits some strong structural layer priors to decompose the layers. The main contributions of this paper are: 1) We propose an effective one-

SUPPORTED BY GRANTS (12&ZD231, 2014BAK09B04, 2013BAK01B05, 61402467).

*Corresponding Author

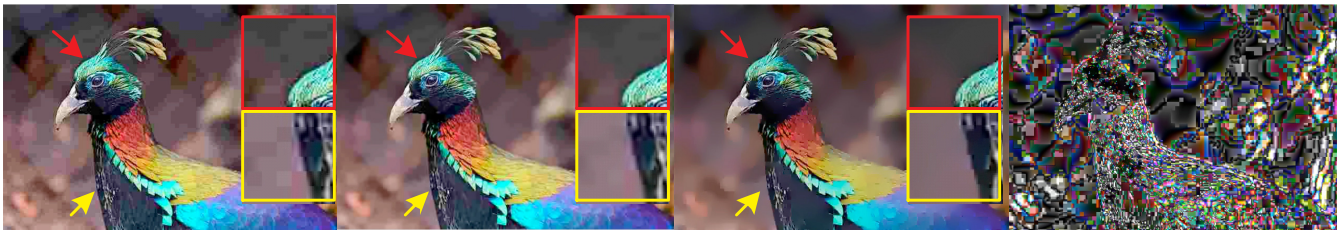

Fig. 1: **Left:** the compressed frame. **Mid-Left:** the recovered intrinsic layer by JAS [3]. **Rest:** the recovered intrinsic $\mathcal{L}_I$ layer and artifact $\mathcal{L}_A$ layers by our proposed DSLP, respectively. Please see the zoomed-in patches for details.

step visual data deblocking method that harnesses two structural layer priors, *i.e.* the *independence* between the gradient fields of the two layers, and the *sparsity* of the gradient field of the intrinsic layer in a unified fashion, and 2) We design a novel Augmented Lagrange Multiplier based algorithm to efficiently and effectively seek the solution of the associated optimization problem.

2. METHODOLOGY

2.1. Notations

Lowercase letters $(a, b, ...)$ mean scalars, bold lowercase letters $(\boldsymbol{a}, \boldsymbol{b}, ...)$ vectors, while bold uppercase letters $(\boldsymbol{A}, \boldsymbol{B}, ...)$ matrices. Bold calligraphic uppercase letters $(\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}}, ...)$ represent high order tensors. $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{D_1 \times D_2 \times \cdots \times D_n}$ denotes an n-order tensor. The Frobenius and ℓ^1 norms of $\boldsymbol{\mathcal{A}}$ are respectively defined as $\|\boldsymbol{\mathcal{A}}\|_F := \sqrt{\sum a^2_{d_1,d_2,...,d_n}}$ and $\|\boldsymbol{\mathcal{A}}\|_1 := \sum |a_{d_1,d_2,...,d_n}|$, while the ℓ^0 norm $\|\boldsymbol{\mathcal{A}}\|_0$ is the number of non-zero elements in $\boldsymbol{\mathcal{A}}$. The inner product of two tensors with identical size is computed as $\langle \boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}} \rangle := \sum (a_{d_1,d_2,...,d_n} \cdot b_{d_1,d_2,...,d_n})$. And $\boldsymbol{\mathcal{A}} \odot \boldsymbol{\mathcal{B}}$ means the Hadamard product of two tensors with same size.

2.2. Problem Formulation

To be general, we employ tensors as the information container. For instance, a gray image is a 2-order tensor, a color image 3-order, while a color video 4-order. Recall that the compressed image or video sequence $\boldsymbol{\mathcal{C}}$ is superimposed by the intrinsic $\mathcal{L}_I$ and artifact $\mathcal{L}_A$ components: $\boldsymbol{\mathcal{C}} = \mathcal{L}_I + \mathcal{L}_A$. From this model, however, we can see that the number of unknowns to be recovered is twice as many as that of the given measurements, which indicates that the problem is highly ill-posed. Therefore, without additional knowledge, the decomposition problem is intractable as it has infinitely many solutions and thus, it is impossible to identify which of these candidate solutions is indeed the "correct" one. To make the problem well-posed, we impose additional structural layer priors on the desired solution for $\mathcal{L}_I$ and $\mathcal{L}_A$. It is well known that natural images or videos are largely piecewise smooth in both spatial and temporal, and the gradient field of intrinsic component is typically sparse. We call this *the gradient sparsity prior.* In addition, the gradient fields of the two layers should be statistically (approximately) uncorrelated. Thus, we note this as *the gradient independence prior.* Furthermore, we observe that the fraction of artifact in pixel values is usually much smaller than that of intrinsic.

Before formulating the problem, we first define the tensor mode-k derivative response and generalized tensor gradient.

Definition 1. *(Tensor Mode-k Derivative Response.) The derivative response of an n-order tensor $\boldsymbol{\mathcal{A}}$ along mode-k ($k \in \{1, 2, ..., n\}$) fibers is defined as:*

$$\mathfrak{R}(\boldsymbol{\mathcal{A}}, k) \in \mathbb{R}^{D_1 \times D_2 \times \cdots \times D_n} := \text{fold}(f_{\frac{\pi}{2}} * \boldsymbol{A}_{[k]}, k),$$

where $f_{\frac{\pi}{2}}$ is the vertical derivative filter and $*$ is the operator of convolution.

Definition 2. *(Generalized Tensor Gradient.) The generalized gradient of an n-order tensor $\boldsymbol{\mathcal{A}}$ is defined as:*

$$\nabla \boldsymbol{\mathcal{A}} := \{\mathfrak{R}(\boldsymbol{\mathcal{A}}, 1), \mathfrak{R}(\boldsymbol{\mathcal{A}}, 2), ..., \mathfrak{R}(\boldsymbol{\mathcal{A}}, n)\},$$

which is analogue to the definition of matrix gradient.

Please notice that, for an image $\in \mathbb{R}^{w \times h \times c}$ (w, h and c are its width, height and color channel, respectively) and a video sequence $\in \mathbb{R}^{w \times h \times c \times t}$ (t is the number of frames), the derivative response across different color channels typically does not contain statistical meaning, which is therefore omitted for the rest of the paper. Furthermore, for clarity, we denote ∇_1 and ∇_2 as the spatial response operators in vertical and horizontal directions respectively, while ∇_3 the temporal response operator.

Based on the stated priors and observation, the desired decomposition $(\mathcal{L}_I, \mathcal{L}_A)$ should minimize the following:

$$\begin{aligned} \underset{\mathcal{L}_I, \mathcal{L}_A}{\text{argmin}} \|\mathcal{L}_A\|_F^2 + \sum_{j=1}^{J} (\alpha \|\nabla_j \mathcal{L}_I\|_0 + \beta \|\nabla_j \mathcal{L}_I \odot \nabla_j \mathcal{L}_A\|_0 \\ + \gamma \|\mathcal{G}_j - \nabla_j \mathcal{L}_I - \nabla_j \mathcal{L}_A\|_F^2) \text{ s.t. } \boldsymbol{\mathcal{C}} = \mathcal{L}_I + \mathcal{L}_A \end{aligned} \tag{1}$$

where α, β and γ are the weights controlling the importances of different terms, and $\mathcal{G}_j := \nabla_j \boldsymbol{\mathcal{C}}$ that can be computed beforehand. J can be either 2 for images or 3 for videos. In

the objective function (1), the first term $\|\boldsymbol{\mathcal{L}}_A\|_F^2$ restricts that the artifact layer is light, which is treated as a Gaussian noise. The second term $\sum_{j=1}^{J}\|\nabla_j\boldsymbol{\mathcal{L}}_I\|_0$ essentially enforces the recovered intrinsic layer to have sparse gradient field. And the remaining two terms constrain the gradient fields of the two layers to be independent of each other. More specifically, the third term $\sum_{j=1}^{J}\|\nabla_j\boldsymbol{\mathcal{L}}_I\odot\nabla_j\boldsymbol{\mathcal{L}}_A\|_0$ penalizes the overlapping of the gradient fields of the two layers, while the fourth $\sum_{j=1}^{J}\|\boldsymbol{\mathcal{G}}_j-\nabla_j\boldsymbol{\mathcal{L}}_I-\nabla_j\boldsymbol{\mathcal{L}}_A\|_F^2$ enforces that, gradients do not appear in the observation should not be groundlessly generated in both the two layers, and existing gradients would also not be gratuitously erased.

It is easy to verify that given an n-order tensor $\boldsymbol{\mathcal{A}}\in\mathbb{R}^{D_1\times\cdots\times D_n}$, there exists a functional matrix $\boldsymbol{F}_{pq}\in\mathbb{R}^{\prod_{i=1}^{n}D_i\times\prod_{i=1}^{n}D_i}$ that satisfies $\mathrm{vec}(\mathrm{unfold}(\nabla_p\boldsymbol{\mathcal{A}},1))=\boldsymbol{F}_{pq}\boldsymbol{a}_{[q]}$, for any $p\in\{1,2,...,n\}$ and $q\in\{1,2,...,n\}$. Thus, the formulation of the problem (1), called DSLP (Deblocking using Structural Layer Priors), can be further simplified, as follows:

$$\begin{aligned}\underset{\boldsymbol{\mathcal{L}}_I,\boldsymbol{\mathcal{L}}_A}{\operatorname{argmin}}\ &\|\boldsymbol{\mathcal{L}}_A\|_F^2+\alpha\|\boldsymbol{Fl}_{I[1]}\|_0+\beta\|\boldsymbol{Fl}_{I[1]}\odot\boldsymbol{Fl}_{A[1]}\|_0\\&+\gamma\|\boldsymbol{g}-\boldsymbol{Fl}_{I[1]}-\boldsymbol{Fl}_{A[1]}\|_F^2\ \text{s.t.}\ \boldsymbol{\mathcal{C}}=\boldsymbol{\mathcal{L}}_I+\boldsymbol{\mathcal{L}}_A,\end{aligned}\tag{2}$$

where $\boldsymbol{F}=[\boldsymbol{F}_{11};\boldsymbol{F}_{21};...;\boldsymbol{F}_{J1}]\in\mathbb{R}^{J\prod_{i=1}^{n}D_i\times\prod_{i=1}^{n}D_i}$, and $\boldsymbol{g}=[\mathrm{vec}(\boldsymbol{\mathcal{G}}_{1[1]});\mathrm{vec}(\boldsymbol{\mathcal{G}}_{2[1]});...;\mathrm{vec}(\boldsymbol{\mathcal{G}}_{J[1]})]\in\mathbb{R}^{J\prod_{i=1}^{n}D_i\times 1}$. For the rest of this paper, we will, for brevity, substitute $\boldsymbol{l}_{I[1]}$ and $\boldsymbol{l}_{A[1]}$ with $\boldsymbol{l}_I$ and $\boldsymbol{l}_A$, respectively.

2.3. Optimization

The objective (2) is difficult to directly optimize due to the non-convexity of the ℓ^0 terms. The convex relaxation for these terms is an effective manner to make the problem tractable. Hence, we replace the ℓ^0 norm with its tightest convex surrogate, namely the ℓ^1 norm. In addition, to adopt ALM-ADM [8] to our problem, we need to make our objective function separable. Thus we introduce two auxiliary variables $\boldsymbol{u}$ and $\boldsymbol{v}$ to replace $\boldsymbol{Fl}_I$ and $\boldsymbol{Fl}_A$, respectively. Accordingly, $\boldsymbol{u}=\boldsymbol{Fl}_I$ and $\boldsymbol{v}=\boldsymbol{Fl}_A$ act as the additional constraints. Naturally, we have:

$$\begin{aligned}\underset{\boldsymbol{\mathcal{L}}_I,\boldsymbol{\mathcal{L}}_A}{\operatorname{argmin}}&\|\boldsymbol{\mathcal{L}}_A\|_F^2+\alpha\|\boldsymbol{u}\|_1+\beta\|\boldsymbol{u}\odot\boldsymbol{v}\|_1+\gamma\|\boldsymbol{g}-\boldsymbol{u}-\boldsymbol{v}\|_F^2\\&\text{s.t.}\ \boldsymbol{\mathcal{C}}=\boldsymbol{\mathcal{L}}_I+\boldsymbol{\mathcal{L}}_A,\ \boldsymbol{u}=\boldsymbol{Fl}_I,\ \boldsymbol{v}=\boldsymbol{Fl}_A.\end{aligned}\tag{3}$$

Converting the constrained minimizing problem (3) to the unconstrained gives the augmented Lagrangian function as:

$$\mathfrak{L}=\begin{cases}\|\boldsymbol{\mathcal{L}}_A\|_F^2+\alpha\|\boldsymbol{u}\|_1+\beta\|\boldsymbol{u}\odot\boldsymbol{v}\|_1\\+\gamma\|\boldsymbol{g}-\boldsymbol{u}-\boldsymbol{v}\|_F^2+\Phi(\boldsymbol{\mathcal{X}},\boldsymbol{\mathcal{C}}-\boldsymbol{\mathcal{L}}_I-\boldsymbol{\mathcal{L}}_A)\\+\Phi(\boldsymbol{y}_1,\boldsymbol{u}-\boldsymbol{Fl}_I)+\Phi(\boldsymbol{y}_2,\boldsymbol{v}-\boldsymbol{Fl}_A),\end{cases}\tag{4}$$

with the definition $\Phi(\boldsymbol{\mathcal{A}},\boldsymbol{\mathcal{B}}):=\frac{\mu}{2}\|\boldsymbol{\mathcal{B}}\|_F^2+\langle\boldsymbol{\mathcal{A}},\boldsymbol{\mathcal{B}}\rangle$, where μ is a positive penalty scalar (monotonically increasing during iteration) and, $\boldsymbol{\mathcal{X}}$, $\boldsymbol{y}_1$ and $\boldsymbol{y}_2$ are the Lagrangian multipliers. Besides the Lagrangian multipliers, there are four variables, including $\boldsymbol{\mathcal{L}}_I$, $\boldsymbol{\mathcal{L}}_A$, $\boldsymbol{u}$ and $\boldsymbol{v}$, to solve. The ALM-ADM solver iteratively updates one variable at a time by fixing the others. Fortunately, each step has a simple closed-form solution, and hence can be computed efficiently. The iterative procedure terminates when $\|\boldsymbol{\mathcal{C}}-\boldsymbol{\mathcal{L}}_I^{(t+1)}-\boldsymbol{\mathcal{L}}_A^{(t+1)}\|_F\le\delta\|\boldsymbol{\mathcal{C}}\|_F$ with $\delta=10^{-7}$ or the maximal number of iterations is reached.

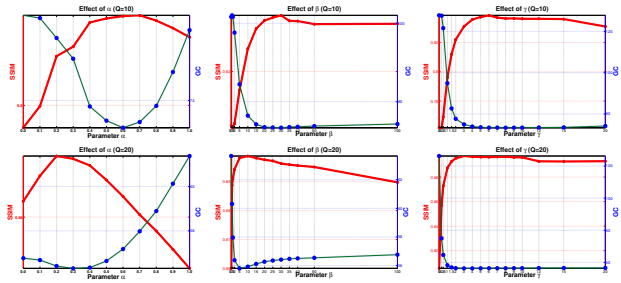

Fig. 2: **Left:** the effect of α with β and γ fixed. **Middle:** the effect of β with α and γ fixed. **Right:** the effect of γ with α and β fixed. **Upper:** the case with JPEG quality 10. **Lower:** the case with JPEG quality 20.

3. EXPERIMENTS

Parameter Effect. Our model involves three free parameters including α, β and γ. We here test the effect of each parameter. The structural similarity (SSIM) metric tries to measure how similar a pair of images are (the deblocked result and its original), which considers three aspects of similarity including luminance, contrast and structure. In addition, we introduce a new metric, gradient consistency (GC), to corporate with SSIM, which is defined as follows:

$$GC(\boldsymbol{\mathcal{A}},\boldsymbol{\mathcal{B}})=\frac{\|\nabla\boldsymbol{\mathcal{A}}-\nabla\boldsymbol{\mathcal{B}}\|_F^2}{\prod_{i=1}^{n}D_i},\tag{5}$$

where $\boldsymbol{\mathcal{A}}$ is the reference and $\boldsymbol{\mathcal{B}}$ the recovery. GC is to see the consistency of gradients of two individuals. Please notice that the higher SSIM the better, while the lower GC the better. Because the dependence of the three parameters is complex, we test them separately. For α, we fix β and γ to 30 and 6, respectively. As can be viewed in Fig. 2, the best α values change from $0.6\sim0.7$ for the case with JPEG quality 10 to $0.2\sim0.3$ for the case with JPEG quality 20 in terms of both SSIM and GC. This result is consistent with the fact that more artifacts require more powerful smoother to eliminate. As for β, we can observe from the second row of Fig. 2 that it performs stably in the range $[15,100]$ for JPEG quality 10 and $[5,100]$ for JPEG quality 20, respectively. Similarly, the parameter γ can achieve high performance when it is set to a relatively large value for both the two cases shown in Fig. 2. For the rest experiments, we will fix β and γ to 30 and 6, respectively.

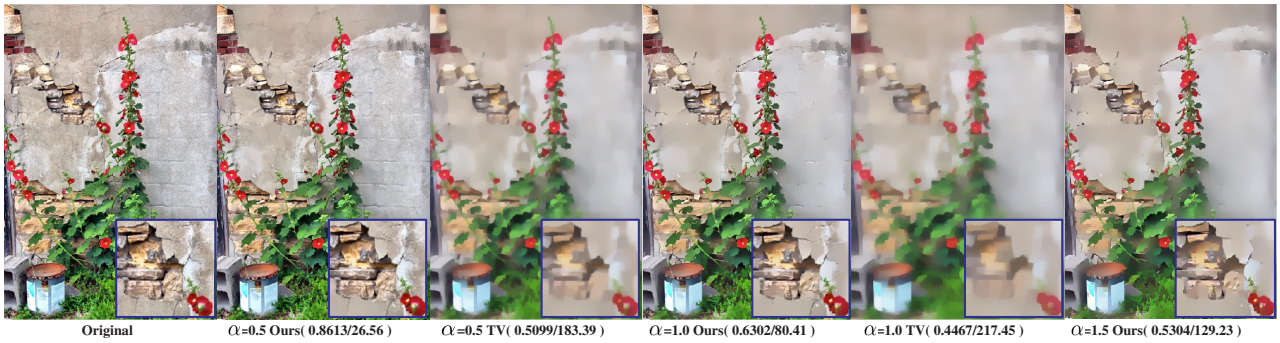

Fig. 3: An illustrative example to reveal the difference between TV model and our method.

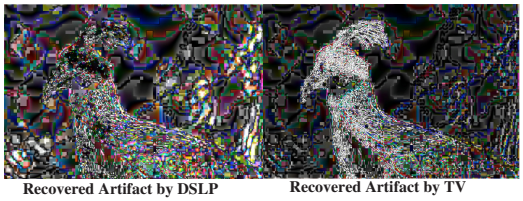

Fig. 4: Visual comparison of recovered artifact between TV and our proposed method.

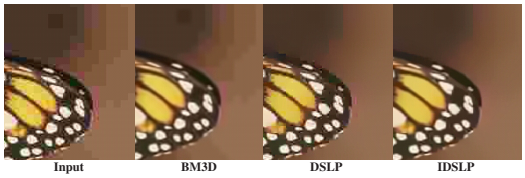

Fig. 5: Illurstration of JPEG compression complication.

Relationship to TV model. From the objective function (3), we can observe that our model reduces to the anisotropic Total Variation (TV) model by disabling the third and fourth terms, say the gradient independence prior. To demonstrate the benefit of the gradient independence prior, we conduct a comparison between TV and our method. To better view the difference, we do not introduce artifacts into the testing. As shown in Fig. 3, bigger α leads to more details smoothed for both TV and DSLP. The difference is that, in terms of visual quality, TV smooths both the high-frequency and low-frequency information, while our DSLP eliminates weak textures but keeps dominant edges. Specifically, when setting α to 1.0, DSLP achieves 0.6302 SSIM and 80.41 GC, which are much better than those of TV, *i.e.* 0.4467 SSIM and 217.45 GC. The results of $\alpha = 0.5$ are analogue. Please note that even increasing α to 1.5, DSLP still can provide very promising result. From the viewpoint of artifact, we further give an example shown in Fig. 4 to see the power of the independence prior. For better view, we amplify the artifact to 10 times of it. As can be seen, TV greatly filters textures with very high false positive ratio (the details of bird body), while DSLP mainly focuses on the block artifacts. The above experimental results reveal the relationship and the difference between TV and DSLP, and demonstrate the advance of DSLP.

IDSLP: Improved DSLP. Let us here revisit the complication of JPEG compression in terms of visual quality. As can be viewed in the first image of Fig. 5 (JPEG Quality 10), there are actually two main issues, say the staircase effect around block boundaries as well as the serration along image edges. The denoising techniques like BM3D [10] can reduce the serration in the frame, but hardly deal with the staircase effect, as shown in the second picture of Fig. 5 (setting $\sigma = 50$). As for DSLP, it is good at cleaning the staircase around block boundaries but likely leaves the serration (see the third picture in Fig. 5, setting $\alpha = 0.6$). Intuitively, we can further improve the visual quality by making use of their respective advantages. The most right result in Fig. 5 demonstrates the effectiveness of such a strategy, which is obtained by firstly executing the denoising technology (in this paper we adopt BM3D, $\sigma = 25$) and then applying DSLP on the denoised version ($\alpha = 0.3$).

Image Deblocking. In this part, we evaluate the performance of our method on image deblocking, compared with the state-of-the art alternatives including a reconstruction based method using Field of Experts (FoE) [9], a local filtering based method via Shape Adaptive DCT (SADCT) [5], a layer decomposition based method for JPEG Artifact Suppression (JAS) [3], a denoising based method BM3D [10], and a Total Variation regularized restoration method (TV) [11].The codes for the competitors are either downloaded from the authors' websites or provided by the authors, their parameters are tuned or set as suggested by the authors for obtaining their best possible results. As for DSLP on image deblocking, only spatial gradients are taken into account, say $\nabla := \{\nabla_1, \nabla_2\}$. In addition, all the codes are implemented in Matlab, which assures the fairness of time cost comparison. We provide the quantitative (SSIM, GC and Time) and qualitative results on several images in Fig. 6, which are compressed by JPEG with quality 10. As can be seen from Fig. 6, FoE, SADCT, JAS and BM3D can only slightly suppress but not thoroughly eliminate the staircase effect under such a compression rate. DSLP is able to eliminate or largely reduce

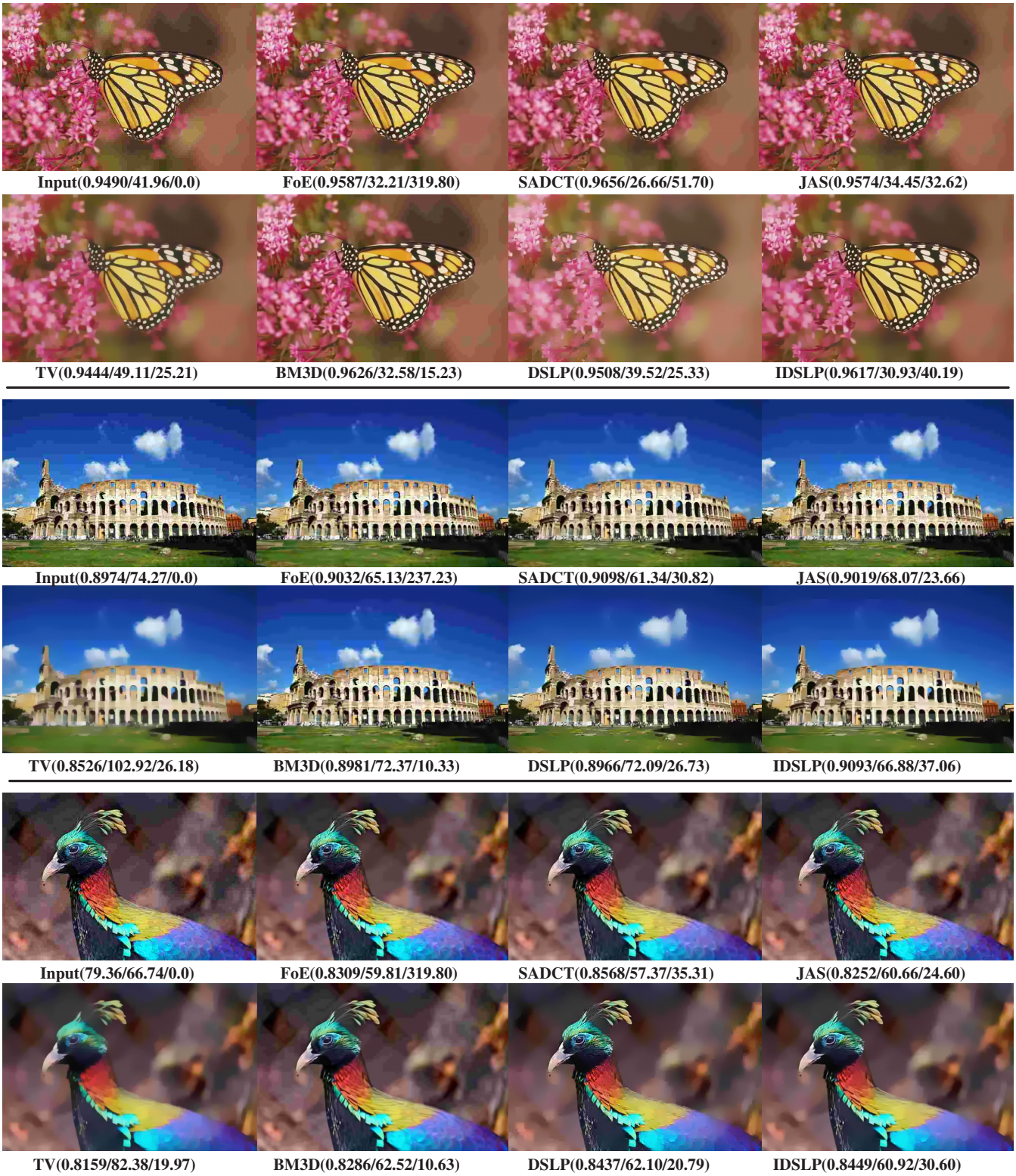

Fig. 6: Performance comparison among FoE [9], SADACT [5], JAS [3], BM3D [10], TV [11], DSLP and IDSLP on image deblocking. Besides the visual results, three quantitative metrics are reported, *i.e.* SSIM/GC/Time(s).

Fig. 7: Visual comparison of video deblocking (16 frames). Two rows correspond to two sample frames.

the staircase, while IDSLP can further mitigate the effect of edge serration. In terms of computational cost, DSLP is superior to SADCT and FoE, and competitive with JAS and TV, but inferior to BM3D. Moreover, IDSLP integrates the denoising and deblocking components, and thus its time cost sums up those of BM3D (for this paper) and DSLP. Due to the limited space and the nature of the deblocking problem, so please see the supplementary material for larger and more results, which are best viewed in original sizes.

Video Deblocking. For this task, we test both spatial only gradients $\nabla := \{\nabla_1, \nabla_2\}$ and spatial-temporal gradients $\nabla := \{\nabla_1, \nabla_2, \nabla_3\}$ for (I)DSLP, which are denoted as (I)DSLP and (I)VDSLP, respectively. This comparison involves VBM3D that is a video extension of BM3D, DSLP, IDSLP and IVDSLP. From Fig. 7, we can see that the problem for BM3D on image deblocking still exists for VBM3D on video deblocking. In other words, the staircase remains (see yellow arrows). DSLP significantly reduces the staircase effect, while IDSLP and IVDSLP further take care of the serration. We note that, compared with IDSLP, IVDSLP slightly excludes some textures (*e.g.* the leaves on the top-right corner, white arrows). This is because the temporal gradient is enforced to be sparse, which would be more helpful for videos with slow motions, but over-smooth the content of videos with sudden or fast motions.

4. CONCLUSION

Artifact separation from images or video sequences is an important, yet severely ill-posed problem. To overcome its difficulty, this paper has shown how to harness two prior structures of the intrinsic and artifact layers. We have formulated the problem in a unified optimization framework and proposed an effective algorithm to seek the solution. The experimental results, compared to the state of the arts, have demonstrated the efficacy of the proposed method in terms of visual quality and simplicity.

5. REFERENCES

[1] "Information technology digital compression and coding of continuous-tone still images requirements and guidelines," Tech. Rep. Tech.Rep.ISO/IEC 10918-1 and ITU T.81, International Telecommunication Union, 1992.

[2] J. Ostermann, J. Bormans, P. List, D. Marpe, M. Narroschke, F. Pereira, T. Stockhammer, and T. Wedi, "Video coding with h.264/avc: Tools, performance, and complexity," *IEEE Circuits and Systems Magazine*, vol. 4, no. 1, pp. 7–28, 2004.

[3] Y. Li, F. Guo, R. Tan, and M. Brown, "A contrast enhancement framework with jpeg artifacts suppression," in *ECCV*, 2014, pp. 174–188.

[4] J. Lim and H. Reeve, "Reducing of blocking effect in image coding," *Journal of Optical Engineering*, vol. 23, pp. 34–37, 1984.

[5] A. Foi, V. Katkovnik, and K. Egiazarian, "Pointwise shape-adaptive dct for high-quality denoising and deblocking of grayscale and color images," *TIP*, vol. 16, no. 5, pp. 1395–1411, 2007.

[6] C. Jung, L. Jiao, H. Qi, and T. Sun, "Image deblocking via sparse representation," *Singal Processing: Image Communication*, vol. 27, no. 6, pp. 663–677, 2012.

[7] D. Sun and C. Liu, "Non-causal temporal prior for video deblocking," in *ECCV*, 2012, pp. 510–523.

[8] Z. Lin, R. Liu, and Z. Su, "Linearized alternating direction method with adaptive penalty for low rank representation," in *NIPS*, 2011, pp. 612–620.

[9] D. Sun and W. Cham, "Postprocessing of low bit-rate block dct coded images based on a fields of experts prior," *TIP*, vol. 16, no. 11, pp. 2743–2751, 2007.

[10] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3d transform-domain collaborative filtering," *TIP*, vol. 16, no. 8, pp. 2080–2095, 2007.

[11] S. Chan, R. Khoshabeh, K. Gibson, P. Gill, and T. Nguyen, "An augmented lagrangian method for total variation video restoration," *TIP*, vol. 20, no. 11, pp. 3097–3111, 2011.