IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received September 24, 2017, accepted November 3, 2017, date of publication November 15, 2017, date of current version December 22, 2017.

Digital Object Identifier 10.1109/ACCESS.2017.2773503

Image Smoothing via Truncated Total Variation

ZEYANG DOU[1], MENGNAN SONG[2], KUN GAO[1], AND ZEQIANG JIANG[3]

[1]Key Laboratory of Photoelectronic Imaging Technology and System, Beijing Institute of Technology, Beijing CO 100081, China
[2]Youli.com, Beijing CO 100081, China
[3]Graduate School, National Defence University of PLA, Beijing CO 100091, China

Corresponding author: Mengnan Song (hdsmncom@163.com)

This work was supported in part by the Natural Science Foundation of Beijing, China, under Grant 4152045 and in part by the National Science Foundation of China under Grant 61527802.

ABSTRACT We present a new regularizer for image smoothing which is particularly effective for diminishing insignificant details, while preserving salient edges. The proposed regularizer relates in spirit to total variation which penalizes all the gradients, while our method just penalizes part of the gradients and leaves the significant edges unchanged. Though the proposed regularizer is a piecewise function, which is hard to optimize, we can unify it to a mathematically sound penalty. The unified penalty term is easy to optimize using recent fast solvers and hard thresholding operation. We show some potential applications of the proposed regularizer, including texture removal and compression artifact restoration. The results show the efficiency of the proposed regularizer.

INDEX TERMS Image smoothing, total variation model, truncated total variation, split Bregman iteration.

I. INTRODUCTION

EDGE preserving image smoothing, which has been a fundamental tool for variety of applications in computer vision and image processing, aims to smooth the details of the image while preserves the significant edges. The image is decomposed into two layers via image smoothing: the piecewise smooth base layer and a detail layer. Such decomposition then can be used for HDR tone mapping [1], [2], texture manipulating [3], flash/no-flash image fusion [4], transfer of photographic look [5], and for other tasks.

Traditionally, smoothing images are done using varieties of filters. The earliest Gaussian smoothing is known to produce halo artifacts near edges. These artifacts may be eliminated by using non-linear edge-preserving smoothing filters such as bilateral filter [6], anisotropic diffusion [12], weighted least squares [7], total variation (TV) [8] and guided filter [9]. Bilateral filtering is widely used due to its simplicity and effectiveness in removing uninterested textures. This filter trades off between details removal and edge preservation [6]. A 1D example is shown in Fig. 1(a). Anisotropic diffusion [12] also aims at suppressing noise while preserving significant structures by introducing an edge-stopping diffusion coefficient. The edge-stopping diffusion coefficient prevents smoothing from crossing strong structures. One example is shown in Fig. 1(b). Farbman *et al.* [7] proposed an edge preserving filter using the weighted least square (WLS) framework. The proposed filter is more flexible compared with previous local filters. Its result is shown in Fig. 1(c). Xu *et al.* [10] proposed the L0 regularizer to control the image smoothness by counting non-zero gradients. Their framework not only eliminates the low-amplitude structures but also slightly sharpens major edges. One example is shown in Fig. 1(d). Later, Xu *et al.* [13] proposed the relative total variation (RTV) to separate the texture patterns from the textured images. Zhang *et al.* [14] proposed the rolling guidance filter (RGF) framework to separate different scale structures. Recently, many learning based filters [15]–[17] appeared, while they require the large amount of data to learn, which may be impossible for some applications, and many of them are based on the fundamental filters such as bilateral filters [16], [17].

TV, which is also an edge-preserving smoothing filter, is widely used due to its simplicity and effectiveness in removing noise-like structures. However, as shown in Fig. 1(e), it penalizes large gradient magnitudes, possibly degrading contrast during smoothing. We try to rescue this problem by establishing a mathematically sound penalty. In this paper, we present a variant of TV, which is called truncated TV, greatly helpful for smoothing the uninteresting structures of the images and retaining the significant edges. Compared to traditional TV, we try to penalize parts of the gradient of the image, assuring that only salient structures in the image

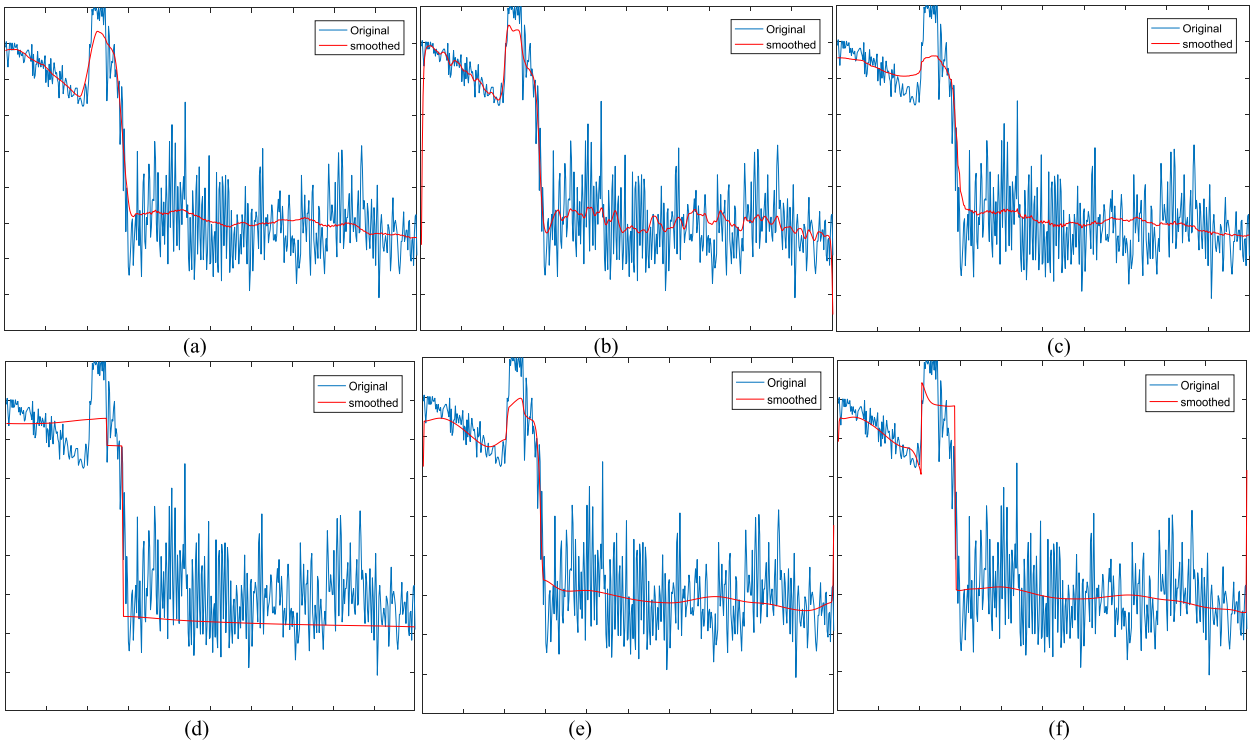


FIGURE 1. Signal captured from a real image, containing both significant edges and abundant details. (a) Result of bilateral filter; (b) Result of anisotropic diffusion; (c) Result of WLS framework; (d) Result of L_0 smoothing; (e) Result of TV; (f) Result of proposed truncated TV.

are preserved. The truncated TV is mathematically established with the high-sparsity-pursuit regularization. Besides, the proposed algorithm can take advantages of the recent fast TV solvers without many modifications, resulting in the same computational efficiency of TV. Although the proposed method is simple, it surprisingly produces the comparable or even better results compared with some specifically designed filters such as RTV. The proposed regularizer is fundamental has many applications, such as compression-artifact degraded clip-art recovery, texture removal and so on.

The rest of the paper is organized as follows. In section II we describe the truncated TV and give the optimization procedure based on the split Bregman framework. Section III shows several applications and gives detailed comparisons with other existing edge preserving smooth techniques. Section IV concludes the paper.

II. TRUNCATED TV

We denote the input image by f and the smoothed result by u. The truncated TV is expressed as

$$U(x) = \int (T(u_x) + T(u_y))dxdy \tag{1}$$

$$T(u) = \begin{cases} |u|| \quad u(x)| < \varepsilon \\ \varepsilon \quad \text{otherwise} \end{cases} \tag{2}$$

where (u_x, u_y) is the gradient of u. The truncated TV only penalizes gradients whose magnitudes are smaller than the threshold ε, while for those whose magnitudes are greater than ε, the proposed penalty doesn't penalize them. Fig 2 shows the shapes of TV and truncated TV. We see that compared with TV, the shape of truncated TV looks like the "truncated" version of TV.

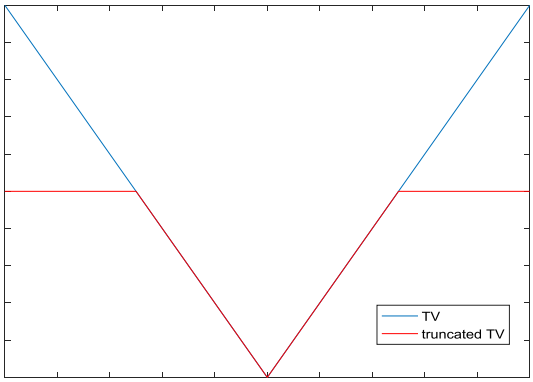


FIGURE 2. Plots of TV and truncated TV.

Truncated TV is non-convex and difficult to optimize due to its "truncated" shape. However, (2) can be re-expressed using L_0 regularization. Taking ε as a parameter, $T(x)$ defined in (2) is equivalent to

$$\phi(u, l) = \min_l \{\varepsilon |l|^0 + |u - l|\}. \tag{3}$$

Where $|\cdot|^0$ is zero power operator, i.e. $|l|^0 = 1$ if $l \neq 0$, else $|l|^0 = 0$. We give the formal proof for the equivalence about two functions.

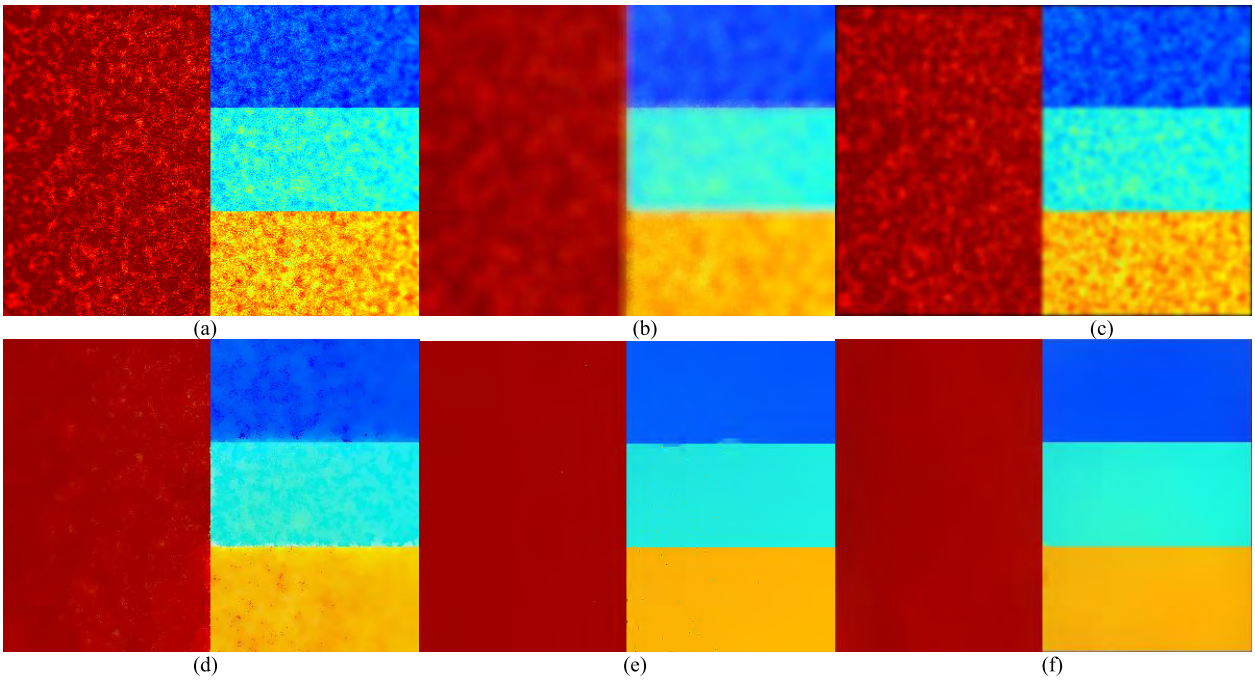


FIGURE 3. Noisy image created by Farman et al. (a) Noisy input; (b) Result of bilateral filter; (c) Result of anisotropic diffusion; (d) Result of WLS framework; (e) Result of L_0 smoothing; (f) Result of proposed method.

Theorem 1: The penalty function (3) is equivalent to the function in (2) given the optimal l.

Proof: If $|u| < \varepsilon$, we compare the output of (3). If $l \neq 0$, then

$$\varepsilon|l|^0 + |u - l| \geq \varepsilon. \tag{4}$$

If $l = 0$,

$$\varepsilon|l|^0 + |u - l| < \varepsilon. \tag{5}$$

So the optimal l is 0.

If $|u| \geq \varepsilon$, similarly, if $l \neq 0$, we have

$$\min : \varepsilon|l|^0 + |u - l| = \varepsilon \tag{6}$$

when $l = u$. If $l = 0$,

$$\varepsilon|l|^0 + |u - l| \geq \varepsilon \tag{7}$$

So the optimal l is u.

From the above discussion, the optimal l of (3) is concluded as follows:

$$l = \begin{cases} u & |u| \geq \varepsilon \\ 0 & |u| < \varepsilon \end{cases} \tag{8}$$

With the optimal l, (3) becomes

$$\phi(u, l) = \begin{cases} \varepsilon & |u| \geq \varepsilon \\ |u| & |u| < \varepsilon \end{cases} \tag{9}$$

End the proof.

Theorem 1 helps us to transform the original non-convex and difficult penalty into an algorithmically practical and effective one. The final objective is

$$\min E(u, l) = \min_{u,l_1,l_2} \{ \int (f-u)^2 dxdy + \alpha \int (\phi(u_x, l_1) + \phi(u_y, l_2)) dxdy \} \tag{10}$$

The first term of the right side of (10) is the data fidelity term, and the second is truncated TV. Fig 1(f) shows 1D smoothing result using truncated TV. Compared with the other state-of-the-art smoothing models, truncated TV makes a good balance between the detail smoothing and the strong edge preserving, while other filters either degrade the strong edges (e.g. bilateral filter, anisotropic diffusion, WLS framework, TV) or smooth the interesting details too much (e.g. L_0 smoothing). We further give a 2D example created by Farbman *et al.* [7] to evaluate and compare smoothing results. The color image showed in Fig. 3 is a piece-wise constant image corrupted with intensive noise. (b) - (f) show the results of four state-of-the-art methods and the proposed model. While other models either corrupt strong edges or leave isolate noise, our model generates the best result shown in (f).

Directly minimizing functional (10) is difficult because it involves L_1 and L_0 penalty terms. We adopt an alternating optimization strategy with split Bregman framework [11]. The key idea is introducing auxiliary variables to expand the original terms and update them alternately.

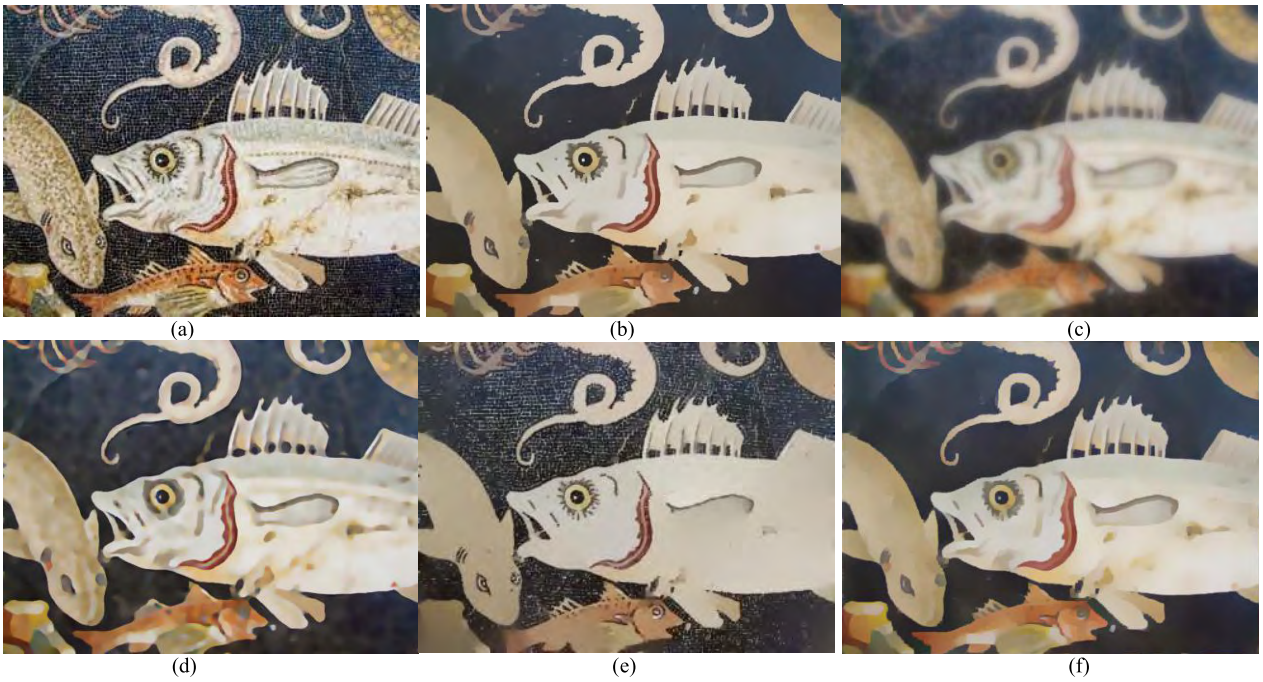


FIGURE 4. Texture removal results comparisons. (a) Input image; (b) RTV; (c) TV; (d) RGF; (e) L_0 smoothing; (f) Proposed method.

We introduce two dual variables b_1 and b_2, corresponding to $u_x - l_1$ and $u_y - l_2$ respectively, and re-express the objective function (10) as follows:

$$\begin{aligned} &E(u, l_1, l_2, b_1, b_2) \\ &= \min \quad \{\int (f-u)^2 dxdy + \alpha\varepsilon \int (|l_1|^0 + |l_2|^0) dxdy \\ &\qquad + \alpha \int (|b_1| + |b_2|) dxdy \qquad (11) \\ &\quad s.t.\ \ b_1 = u_x - l_1, \quad b_2 = u_y - l_2. \qquad (12) \end{aligned}$$

By using Lagrangian multipliers, (11) can be converted to an unconstraint problem:

$$\begin{aligned} &E(u, l_1, l_2, b_1, b_2) \\ &= \min\{\int (f-u)^2 dxdy + \alpha\varepsilon \int (|l_1|^0 + |l_2|^0) dxdy \\ &\quad + \alpha \int (|b_1| + |b_2|) dxdy + \lambda \int (b_1 - u_x + l_1)^2 dxdy \\ &\quad + \lambda \int (b_2 - u_y + l_2)^2 dxdy\} \qquad (13) \end{aligned}$$

By introducing Bregman distance, (13) becomes:

$$\begin{aligned} &E(u, l_1, l_2, b_1, b_2, t_1, t_2) \\ &= \min\{\int (f-u)^2 dxdy + \alpha\varepsilon \int (|l_1|^0 + |l_2|^0) dxdy \\ &\quad + \alpha \int (|b_1| + |b_2|) dxdy + \lambda \int (b_1 - u_x + l_1 - t_1)^2 dxdy \\ &\quad + \lambda \int (b_2 - u_y + l_2 - t_2)^2 dxdy\} \qquad (14) \end{aligned}$$

The above joint minimizing problem can be solved alternately by decoupling it into several subproblems, described as follows:

(1) Calculate u subproblem with fixed $l_1, l_2, b_1, b_2, t_1, t_2$

$$\begin{aligned} u^{i+1} = \min_u \{\int (f-u)^2 dxdy + \lambda \int ((b_1^i - u_x + l_1^i - t_1^i)^2 \\ + (b_2^i - u_y + l_2^i - t_2^i)^2) dxdy \qquad (15) \end{aligned}$$

We derive the Euler-Lagrange equation as follows:

$$f - u + \lambda\left(\frac{\partial(b_1^i + l_1^i - t_1^i)}{\partial x} + \frac{\partial(b_2^i + l_2^i - t_2^i)}{\partial y} - \Delta u\right) = 0 \qquad (16)$$

where Δ is the Laplace operator. Equation (16) can be efficient solved by using Gauss-Seidel iteration algorithm or FFT operator.

(2) Calculate b_1 and b_2 with fixed u, l_1, l_2, t_1, t_2

The unique minimizer of this subproblem can be obtained by applying the shrinkage operator:

$$b_1^{i+1} = shrink(u_x^{i+1} - l_1^i + t_1^i, \frac{\alpha}{\lambda}) \qquad (17)$$

$$b_2^{i+1} = shrink(u_y^{i+1} - l_2^i + t_2^i, \frac{\alpha}{\lambda}) \qquad (18)$$

where

$$shrink(x, \alpha) = \frac{x}{|x|} \max(x - \alpha, 0) \qquad (19)$$

(3) Update t_1, t_2 with fixed u, l_1, l_2, b_1, b_2

$$t_1^{i+1} = u_x^{i+1} + t_1^i - l_1^i - b_1^{i+1} \qquad (20)$$

$$t_2^{i+1} = u_y^{i+1} + t_2^i - l_2^i - b_2^{i+1} \qquad (21)$$

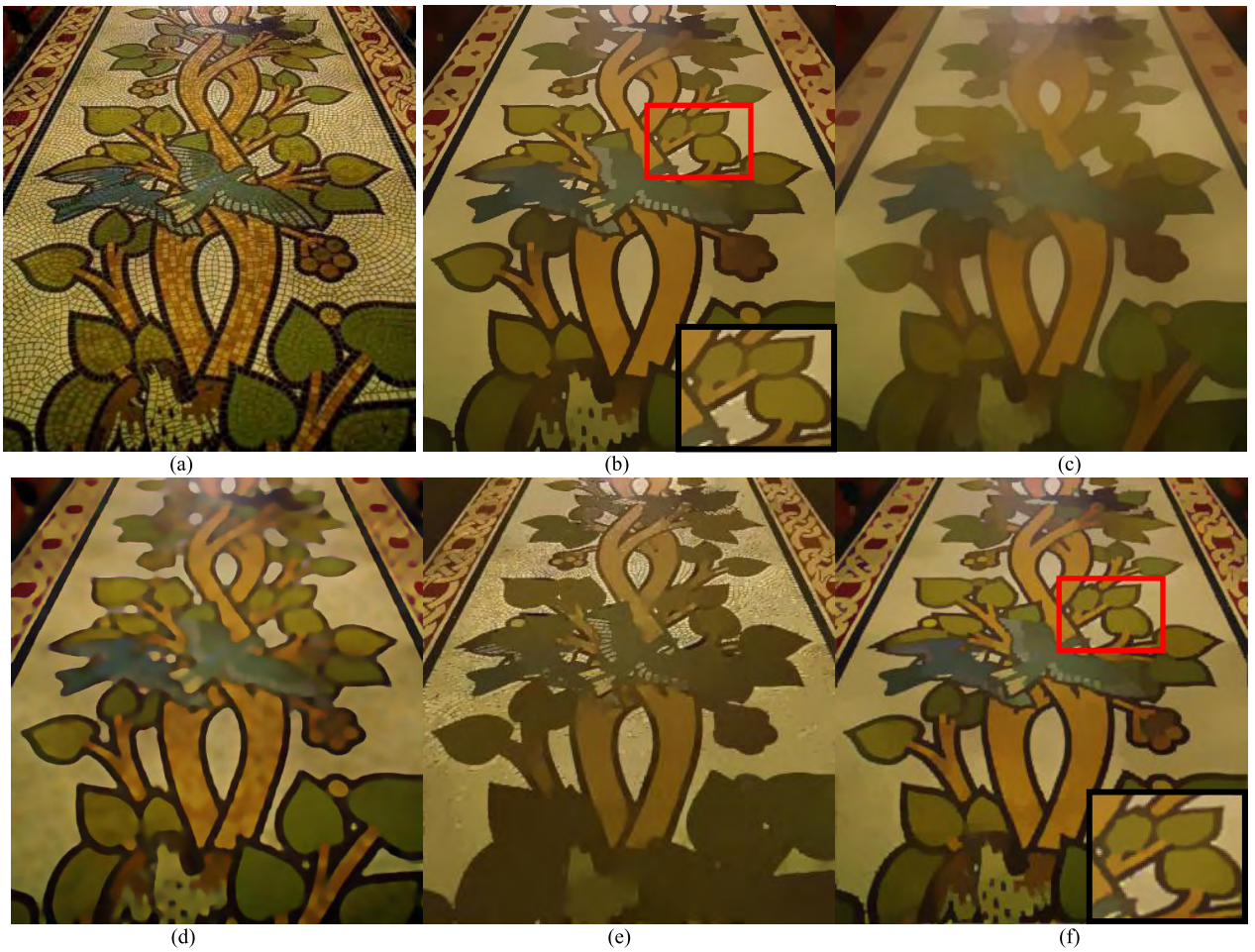


FIGURE 5. Texture removal results comparisons. (a) Input image; (b) RTV; (c) TV; (d) RGF; (e) L_0 smoothing; (f) Proposed method.

(4) Update l_1, l_2 with fixed u, t_1, t_2, b_1, b_2

$$l_1^{i+1}, l_2^{i+1} = \underset{l_1,l_2}{\operatorname{argmin}}\{\alpha\varepsilon \int (|l_1|^0+|l_2|^0)dxdy + \lambda \int ((l_1-u_x^{i+1}+b_1^{i+1}-t_1^{i+1})^2 + (l_2-u_y^{i+1}+b_2^{i+1}-t_2^{i+1})^2)dxdy\} \quad (22)$$

Similar to theorem 1, we see that the solution of this subproblem is:

$$l_1^{i+1} = \begin{cases} t_1^{i+1}-b_1^{i+1}+vu_x^{i+1} & \text{if}|t_1^{i+1}-b_1^{i+1}+u_x^{i+1}| < \sqrt{\frac{\alpha\varepsilon}{\lambda}} \\ 0 \quad \text{otherwise} \end{cases} \quad (23)$$

$$l_2^{i+1} = \begin{cases} t_2^{i+1}-b_2^{i+1}+u_y^{i+1} & \text{if}|t_2^{i+1}-b_2^{i+1}+u_y^{i+1}| < \sqrt{\frac{\alpha\varepsilon}{\lambda}} \\ 0 \quad \text{otherwise} \end{cases} \quad (24)$$

Now we summarize the whole algorithm as algorithm 1.

From the algorithm 1, we can see that all the subproblems have closed form solutions, making the algorithm very fast.

Algorithm 1 Algorithm 1

Input: $u = f$; $b_1 = b_2 = t_1 = t_2 = l_1 = l_2 = 0$
For i from 1 to N:
  Update u by solving (16);
  Update b_1, b_2 using (17) and (18);
  Update t_1, t_2 using (20) and (21);
  Update l_1, l_2 using (24) and (24);
End
Output: u

Compared to recent fast TV solver, the proposed algorithm only adds two dual variables, l_1 and l_2, and two hard threshold operations. Table 1 lists the computation cost of different algorithms. From table 1, we see that our algorithm is highly competitive with the other state-of-the-arts. Note that the proposed algorithm is easy to be parallelized, further improving the performance of the algorithm.

III. APPLICATIONS

Due to its fundamentality, the truncated TV can be applied to many applications. In this paper, we show four applications,

FIGURE 6. Cartoon JPG artifact removal (a) A JPG compressed image with artifacts around edges; (b) Our restoration result; (c) Close-up of the input image; (d) Close-up of RTV; (e) Close-up of TV; (f) Close-up of WLS framework; (g) Close-up of L0 smoothing; (h) Close-up of the proposed.

FIGURE 7. Base-detail separation and boosting results. (a) Input image. (b), (d) The base layers using different parameters; (c), (e) The corresponding detail boosting results of (b) and (d).

TABLE 1. Computation cost comparisons

Algorithm	Image size	Time (Seconds)
Bilateral filter	500	5.567
TV	500	2.512
WLS framework	500	2.875
L0 smoothing	500	1.982
Proposed	500	2.531

TABLE 2. SNR Comparisons of different methods.

Image	RTV	TV	WLS	L_0 smoothing	Truncated TV
1	110.32	110.23	99.85	113.53	116.63
2	106.56	104.63	97.32	107.32	109.48
3	100.65	99.68	87.35	102.53	104.16

including texture removal, compression artifact restoration, detail boosting and non-photorealistic abstraction. Four state-of-the-art methods, i.e. relative TV (RTV), TV, Rolling guidance filter (RGF) and L_0 smoothing, are compared.

A. TEXTURE REMOVAL

Many natural scenes and man-made arts contain texture. Texture extraction by the computer is challenging and required in the most of photo editing software when dealing with textures. We compare our method with the other related smoothing filters on texture removal. Figure 4(a) and 5(a) show the mosaic fish and floor images, and Fig. 4(b-f) and 5(b-f) show the results obtained by the state-of-the-art methods and the proposed algorithm. We see that both RTV and truncated TV can keep the good balance between texture removal and preserving the strong structures, while the other methods either blur the consequential structures in the image or leave textures. However, truncated TV leaves fewer unwished textures and produces less halo effect than RTV. These two examples validate the efficiency of the regularizer.

B. COMPRESSION ARTIFACT RESTORATION

Compression artifacts are inevitable in the image format conversions. Our smoothing method also suits for this kind of artifact removal due to its structure preserving property. We compare our method with the other four methods. One comparison is shown in figure 6. From the close-ups in fig. 6, we can see that while all the methods remove the compression artifacts, our method leaves the most abundant structures, e.g. the blush on the cheek of the rabbit and the details of the ears. We further quantitatively validate our method using simulation images. Three images, which are in png format and not shown in the paper, are translated to jpg format images using matlab function imwrite. Then we use the four state-of-the-art methods and the proposed method to eliminate compression artifacts and calculate SNR. Fig 6 shows the SNR comparisons. We see that our method produces the best results.

C. DETAIL BOOSTING

Detail boosting is widely used in recent image processing software. Our method can separate the detail layer from the base layer in controllable degrees by varying the parameter ε, as shown in Fig. 7 (b) and (d). The boosting results are obtained by directly magnifying the detail layer and adding back to the base layer. Figure 7 (c) and (e) show the boosting results of (b) and (d) respectively.

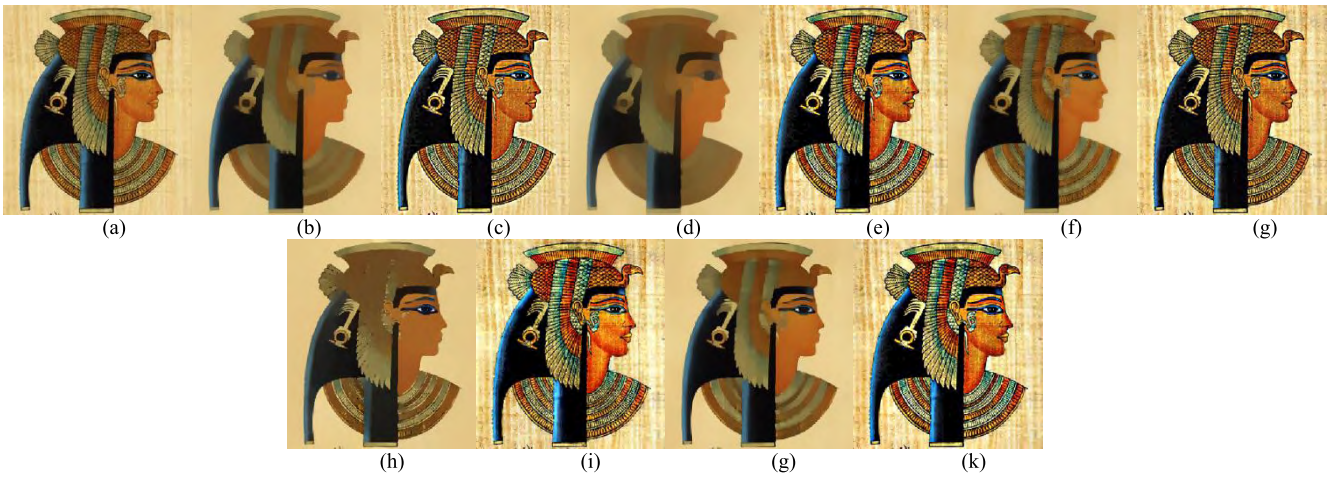


FIGURE 8. Base-detail separation and detail amplification. (a) Input image; (b), (d), (f), (h), (g) are base layers obtained by RTV, TV, WLS framework, L0 smoothing and proposed; (c), (e), (g), (i), (k) represent the corresponding detail amplification results.

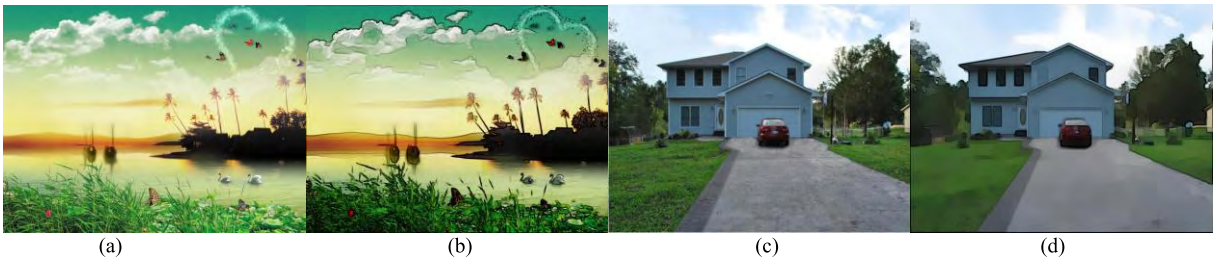


FIGURE 9. Non-Photorealistic Abstraction. (a), (c) Input image; (b), (d) non-photorealistic abstraction using our method.

Figure 8 gives comparisons with the other four methods. The painting has rich textures, which are hard to completely separate from the base layer. Our base layer result contains nearly no texture details, and strong edges are no blurred, while other results either smooth the salient structures or leave abundant details.

D. NON-PHOTOREALISTIC ABSTRACTION

The proposed truncated TV also fits non-photorealistic abstraction with simultaneous detail eliminating and edge emphasizing. Two main steps are involved, first, image smoothing by the edge preserving filter, and then line extraction by edge detector. The extracted lines are enhanced and are composed back to enhance the visual distinctiveness of different regions. Fig. 9 shows two examples using our method.

IV. CONCLUSIONS

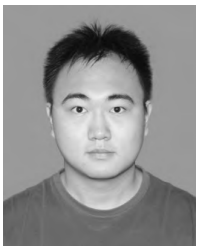
We present a variant of TV, which is called truncated TV, greatly helpful for smoothing the uninterested structures of the image and retaining the significant edges. Compared to traditional TV, the proposed truncated TV tries to penalize parts of the gradient of the image, assuring that only salient structures in the image is preserved. The truncated TV is mathematically established with a high-sparsity-pursuit regularization. Besides, the proposed algorithm can take advantages of the recent fast TV solvers without many modifications, resulting in the competitive computational efficiency of TV. The framework is general, thus it can be applied to many fields, such as compression-artifact recovery, texture removal and manipulating and so on.

Though we show only four potential applications in this paper, the proposed method can be applied in other fields, such as HDR tone mapping, inpainting, restoration, segmentation, blind deconvolution and so on. It can be further designed as a scale related or learning based filter as well. We leave these ideas as the future work.

REFERENCES

[1] J. Tumblin and G. Turk, "LCIS: A boundary hierarchy for detail-preserving contrast reduction," in *Proc. CCGIT*, 1999, pp. 83–90.
[2] F. Durand and J. Dorsey, "Fast bilateral filtering for the display of high-dynamic-range images," *ACM Trans. Graph.*, vol. 21, no. 3, pp. 257–266, 2002.
[3] E. A. Khan, E. Reinhard, R. W. Fleming, and H. H. Bülthoff, "Image-based material editing," *ACM Trans. Graph.*, vol. 25, no. 3, pp. 654–663, 2006.
[4] G. Petschnigg, R. Szeliski, M. Agrawala, M. Cohen, H. Hoppe, and K. Toyama, "Digital photography with flash and no-flash image pairs," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 664–672, 2004.
[5] S. Bae, S. Paris, and F. Durand, "Two-scale tone management for photographic look," *ACM Trans. Graph.*, vol. 25, no. 3, pp. 637–645, 2006.
[6] M. Elad, "On the origin of the bilateral filter and ways to improve it," *IEEE Trans. Image Process.*, vol. 11, no. 10, pp. 1141–1151, Oct. 2002.

[7] Z. Farbman, R. Fattal, D. Lischinski, and R. Szeliski, "Edge-preserving decompositions for multi-scale tone and detail manipulation," *ACM Trans. Graph.*, vol. 27, no. 3, 2008, Art. no. 67.
[8] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Phys. D, Nonlinear Phenomena*, vol. 60, nos. 1–4, pp. 259–268, 1992.
[9] K. He, J. Sun, and X. Tang, "Guided image filtering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 6, pp. 1397–1409, Jun. 2013.
[10] L. Xu, C. Lu, Y. Xu, and J. Jia, "Image smoothing via L_0 gradient minimization," *ACM Trans. Graph.*, vol. 30, no. 6, 2011, Art. no. 174.
[11] T. Goldstein and S. Osher, "The split Bregman method for L1-regularized problems," *SIAM J. Imag. Sci.*, vol. 2, no. 2, pp. 323–343, 2009.
[12] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 7, pp. 629–639, Jul. 1990.
[13] L. Xu, Q. Yan, Y. Xia, and J. Jia, "Structure extraction from texture via relative total variation," *ACM Trans. Graph.*, vol. 31, no. 6, 2012, Art. no. 139.
[14] Q. Zhang, X. Shen, L. Xu, and J. Jia, "Rolling guidance filter," in *Proc. ECCV*, 2014, pp. 815–830.
[15] Y. Li, J.-B. Huang, N. Ahuja, and M.-H. Yang, "Deep joint image filtering," in *Proc. ECCV*, 2016, pp. 154–169.
[16] M. Gharbi, J. Chen, J. T. Barron, S. W. Hasinoff, and F. Durand, "Deep bilateral learning for real-time image enhancement," *ACM Trans. Graph.*, vol. 36, no. 4, 2017, Art. no. 118.
[17] J. T. Barron and B. Poole, "The fast bilateral solver," in *Proc. ECCV*, 2016, pp. 617–632.



ZEYANG DOU received the B.E. degree in mathematics from Baoding University, China, in 2012, and the M.S. degree in computational mathematics from the Communication University of China, China, in 2016. He is currently pursuing the Ph.D. degree with the Beijing Institute of Technology. His research interests include image processing, machine learning, and deep learning.

MENGNAN SONG received the B.E. degree in electronic science and technology from the South China University of Technology, China, in 2013, and the M.S. degree in computer science from Waseda University, Japan, in 2015. Since 2015, he has been involved in software and algorithm development. His research interests include distributed architecture and machine learning.

KUN GAO received the B.A. degree in electrical engineering and the Ph.D. degree in instrument science and engineering from Zhejiang University, China, in 1995 and 2002, respectively. From 2002 to 2004, he was a Post-Doctoral Fellow with Tsinghua University, China. Since 2005, he has been with the Beijing Institute of Technology, China, involved in infrared technology and real-time image processing. He is a member of the Optical Society of China.

ZEQIANG JIANG is currently pursuing the master's degree with the National Defence University of PLA, China. His research interests include optimization and machine learning.