

POCS-Based Blocking Artifacts Suppression Using a Smoothness Constraint Set With Explicit Region Modeling

Alan Wee-Chung Liew, *Member, IEEE*, Hong Yan, *Senior Member, IEEE*, and Ngai-Fong Law

Abstract—It is well known that low bit rate block-based discrete cosine transform coded image exhibits visually annoying coding artifacts. In this paper, we proposed a projection onto convex sets (PCOS)-based deblocking algorithm using a novel region smoothness constraint set for graphic images containing objects with smooth regions. The smoothness constraint set is obtained by an explicit modeling of smooth regions in the image using a spatially adaptive thin-plate spline. In contrast to most deblocking algorithms which enforce smoothness just around the 8×8 block boundaries, our algorithm enforces smoothness in regions which could possibly span several blocks. We showed that convergence of our algorithm could be reached within one iteration. The performance of the proposed algorithm is evaluated visually and quantitatively in term of peak signal-to-noise ratios and the mean squared difference of slope metric, which measures the impact of the blocking effects, for several graphic images. The results show that our algorithm can effectively suppress blockiness in smooth regions while still preserving the sharpness of object edges.

Index Terms—Blocking artifacts, image deblocking, projection onto convex sets (POCS), region smoothness constraint, smooth surface modeling, thin-plate spline.

I. INTRODUCTION

IN BLOCK-BASED discrete cosine transform (BDCT) coding [1], the DCT is applied to each 8×8 nonoverlapping image blocks, followed by quantization and variable length coding. Since each block is independently treated, correlation among spatially adjacent blocks is not taken into account in coding, which results in block discontinuities along the block boundaries in the decoded image when two low-frequency DCT coefficients of similar values in adjacent blocks are quantized into different quantization bins.

Various post-processing methods have been proposed to remove the blocking artifacts so as to improve the visual quality of the decoded image [2]–[11], [18]–[20]. Among them, one popular class of post-processing algorithms is based on the theory of projection onto convex sets (POCS) [4]–[6], [11].

Manuscript received September 30 2003; revised March 27, 2004 and August 9, 2004. This work was supported in part by Hong Kong Research Grant Council (RGC) Project 9 040 516. This paper was recommended by Associate Editor S. U. Lee.

A. W.-C. Liew was with the Department of Computer Engineering and Information Technology, City University of Hong Kong, Kowloon Tong, Hong Kong. He is now with the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Shatin, Hong Kong (e-mail: wcliew@cse.cuhk.edu.hk).

H. Yan is with the Department of Computer Engineering and Information Technology, City University of Hong Kong, Kowloon Tong, Hong Kong and also with the School of Electrical and Information Engineering, University of Sydney, Sydney NSW 2006, Australia.

N. F. Law is with the Centre for Multimedia Signal Processing, Department of Electronic and Information Engineering, The Hong Kong Polytechnic University, Hong Kong.

Digital Object Identifier 10.1109/TCSVT.2005.848303

POCS methods use iterative technique in nature and may involve substantial computation time. Another major approach for image deblocking involves spatially adaptive postfiltering [7]–[9]. Methods in this category are noniterative in nature. Essentially, block discontinuities are examined, and depending on the pixel smoothness condition on the block boundary, adaptive filtering is applied. However, at very low bit rate, block discontinuities cannot be completely eliminated by spatially adaptive postfiltering.

Most deblocking algorithms are developed for general images. By exploiting the properties within special class of images, deblocking performance can be further improved. Recently, we have developed a deblocking algorithm that has excellent performance on a class of images that are assumed to consist of uniform regions [11]. However, many graphic images contain smoothly varying regions and therefore a method to model the smoothness is necessary in order to perform adequate deblocking on such class of images.

In this paper, we describe a POCS-based deblocking algorithm employing a novel smoothness constraint set to handle images with smooth regions. Specifically, we propose a smoothness constraint set that involves explicit modeling of smooth regions in the image. Since the smoothness constraint is enforced over a region (which could span several blocks) instead of just around the 8×8 block boundaries like in most deblocking algorithms, better suppression of blockiness can be achieved. This paper is organized as follows. Section II gives an overview of the mathematical background for thin-plate spline surface modeling and the theory of POCS for image reconstruction. Section II-A gives details about the construction of a smooth, spatially adaptive thin-plate spline surface from the BDCT coded image. Section II-B describes the formulation of the constraint sets for the region smoothness constraint, quantization constraint and range constraint. The performance of the algorithm is evaluated both visually and quantitatively in terms of PSNR and the recently introduced mean squared difference of slope (MSDS) metric in Section IV. Finally, Section V concludes the paper.

II. MATHEMATICAL BACKGROUNDS

A. Thin-Plate Spline Surface Modeling

A thin-plate spline $f(x, y)$ [12], [13], [21], [22] is a smooth function which interpolates a surface at the landmark points $z_i(x, y)$. This function behaves like a thin metal plate and minimizes the total bending energy given by

$$E = \iint_{R^2} \left(\frac{\partial^2 f}{\partial x^2} \right)^2 + 2 \left(\frac{\partial^2 f}{\partial x \partial y} \right)^2 + \left(\frac{\partial^2 f}{\partial y^2} \right)^2 dx dy \quad (1)$$

over the class of such interpolants. Given a set of points $\{z_i(x, y)\}_{i=1\dots m}$, the thin-plate spline takes the form

$$f(x, y) = \sum_{i=1}^m a_i K_i(x, y) + a_{m+1} + a_{m+2}x + a_{m+3}y \quad (2)$$

where

$$K_i(x, y) = r_i^2(x, y) \log r_i^2(x, y) \quad (3)$$

$$r_i(x, y) = \sqrt{(x - x_i)^2 + (y - y_i)^2} \quad (4)$$

and (x_i, y_i) are the x and y coordinates of the point z_i . The summation terms at the right-hand side of (2) are the fundamental solutions of the biharmonic equation, which minimizes the total bending energy. The last three terms describe the lifting and tilting of the spline surface.

Given data point $\{z_i(x_i, y_i)\}_{i=1\dots m}$, the problem at hand is to obtain a best-fit surface $f(x, y)$ with adequate smoothness by minimizing the following regularized cost function with respect to the spline coefficients $a_i, i = 1, \dots, m + 3$ in (2)

$$c(f) = \sum_{i=1}^m [z_i - f(x_i, y_i)]^2 + \lambda E \quad (5)$$

where λ is a scalar that controls the smoothness of the surface. If we ignore λ for the moment (i.e., set $\lambda = 0$), then to solve for the $m + 3$ spline coefficients a_i , the interpolating conditions $f(x_i, y_i) = z_i$ give m linear equations, the remaining three equations are obtained by imposing the orthogonality conditions on the m coefficients

$$\sum_{i=1}^m a_i = 0, \quad \sum_{i=1}^m a_i x_i = 0, \quad \sum_{i=1}^m a_i y_i = 0 \quad (6)$$

In vector-matrix notation, the spline coefficients $a_i, i = 1, \dots, m + 3$, in (2) are then determined as the solution of the linear system

$$Aa = z \quad (7)$$

where

$$A = \begin{bmatrix} K & P \\ P^T & 0 \end{bmatrix}, \quad a = \begin{bmatrix} a_1 \\ \vdots \\ a_m \\ a_{m+1} \\ a_{m+2} \\ a_{m+3} \end{bmatrix}, \quad z = \begin{bmatrix} z_1 \\ \vdots \\ z_m \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (8)$$

$$K = \begin{bmatrix} K_1(x_1, y_1) & \cdots & K_m(x_1, y_1) \\ \vdots & & \vdots \\ K_1(x_m, y_m) & \cdots & K_m(x_m, y_m) \end{bmatrix} \quad (9)$$

$$P = \begin{bmatrix} 1 & x_1 & y_1 \\ \vdots & \vdots & \vdots \\ 1 & x_m & y_m \end{bmatrix}.$$

When λ is present, the coefficients are again obtained by solving (7) but with K replaced by $K + \lambda I$, where I is the $m \times m$ identity matrix [21], [22].

B. POCS Image Reconstruction

In POCS-based image reconstruction [14]–[17], every known property of the original image f can be formulated as a corresponding convex set. The original image is then assumed to lie in the intersection of these convex sets, i.e.,

$$f \in C_0 = \bigcap_{i=1}^m C_i. \quad (10)$$

Since everything we know about the original image is captured in C_0 , the problem of reconstructing an image from its m properties is equivalent to that of finding an element in C_0 . If the projection operator P_i projecting onto the convex set C_i is realizable for $i = 1, 2, \dots, m$, the problem is recursively solvable. The rule that assigns to every $x \in H$ its nearest neighbor $f \in C_i$, i.e., $\|x - P_i x\| = \min_{f \in C_i} \|x - f\|$ defines the nonlinear projection operator $P_i : H \rightarrow C_i$ unambiguously. For the composite operator $T = P_m P_{m-1} \dots P_1$, the recursion $x_n = T^n x$ converges to an element in C_0 as $n = 0 \rightarrow \infty$, for any starting point $x \notin C_0$. The POCS framework ensures that convergence is monotonic and the final solution is optimal mathematically. The optimality here is in the sense that the final solution satisfies all known constraints about the original image. Note that optimality does not ensure uniqueness of the solution, since every solution in the intersection C_0 are equivalent mathematically.

III. DEBLOCKING ALGORITHM

A. Smooth Spline Surface Construction

A thin-plate spline surface for the entire image is obtained by concatenating together the local thin-plate spline surfaces, which are computed over each 8×8 block. Each 8×8 block can be smoothed adaptively by using different λ in (5), depending on the high frequency content of the block. To blend the local surfaces smoothly, the local surface fitting is done over a larger block, i.e., over a 16×16 block. Then only the fitted surface in the center 8×8 block is retained.

We categorize the degree of smoothness of each 8×8 image block into one of 4 classes, i.e., from 1 (least smooth) to 4 (smoothest). This is done as follows. First, the standard deviations for every 16×6 blocks, *STD16*, and 8×8 blocks, *STD8*, are determined. Then, each block is classified into one of the classes according to the following rule:

$$\begin{aligned} &\text{If } \text{STD } 8 \geq 15, \text{ class} = 1 \\ &\text{Else-if } \text{STD } 16 \leq 10, \text{ class} = 4 \\ &\text{Else-if } \text{STD } 16 \leq 15, \text{ class} = 3 \\ &\text{Else class} = 2. \end{aligned} \quad (11)$$

Instead of fitting the spline surface over all image pixels, the surface is fitted using the data points derived from the compressed image. Depending on the degree of smoothness, the data point configurations for an 8×8 block shown in Fig. 1 are adopted. The data points inside the image block are sampled directly from the compressed image, whereas the data points along block boundary are obtained by 3-point averaging the corresponding pixels along the block boundary to smooth out block discontinuities.

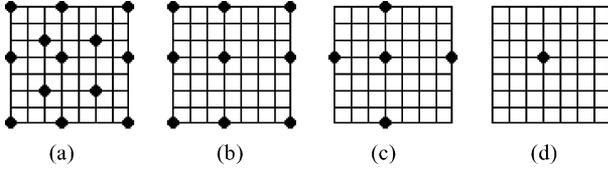


Fig. 1. Data point configuration for (a) class = 1, (b) class = 2, (c) class = 3, and (d) class = 4.

The discontinuities along block boundaries could be due to real edges. In [10], we showed that the minimum block discontinuities, Δ_{\min} , between adjacent blocks is given by $Q(0,0)/8$, where $Q(0,0)$ is the dc entry in the quantization table, and the largest block discontinuity due to quantization is equal to $2\Delta_{\min}$ for a ramp signal. Based on such observation, we flag the discontinuities along block boundaries as real edges when their magnitude is greater than $2\Delta_{\min}$. The image pixels along these real edges are kept as data points without averaging.

To preserve the sharpness of real edges within the block, the gradient magnitude, $M(x,y) = \sqrt{|f_x(x,y)|^2 + |f_y(x,y)|^2}$, and gradient angle, $A(x,y) = \arctan f_y(x,y)/f_x(x,y)$, of the compressed image f are computed, where $f_x(x,y)$ and $f_y(x,y)$ are the partial derivatives of f along the x and y direction, respectively. Then, the image pixels at the maxima of $M(x,y)$ in the direction given by $A(x,y)$, are flagged as real edges and are retained as data points.

The smoothness parameter λ is set according to the smoothness class the image block is assigned to. From experimentation, λ is empirically set to 1, 10, 50, 100 for smoothness class equals to 1, 2, 3, and 4, respectively. Using the data points obtained, the local thin-plate spline surface for each image block is computed by solving (7) and (2), and the spline surface for the whole image is obtained by concatenating these local patches together.

B. Projection Operators

The region smoothness constraint is applied to each of the 8×8 block. Let S_k be the spline surface for the block k and let ε_k be the error bound associated with block k . The region smoothness constraint set for block k is defined as

$$C_s^k = \{f_k : \|f_k - S_k\| \leq \varepsilon_k\}. \quad (12)$$

The set C_s^k can be viewed geometrically as a closed sphere, with a radius of ε_k , and with its center located at point S_k , in a 64-dimension space. It is obvious that the set C_s^k is convex and closed. From this geometrical viewpoint, it is easy to define the nearest projection of any point not in C_s^k onto the surface of this hyper-sphere. Let $g_k \notin C_s^k$ and $f_k \in C_s^k$, where f_k is the nearest projection of g_k onto C_s^k . Define the ratio $\eta = \varepsilon_k / \|g_k - S_k\|$, then, by geometric argument, it can be easily shown that

$$f_k = \eta g_k + (1 - \eta) S_k. \quad (13)$$

Therefore, the projection of g_k onto C_s^k , i.e., $P_S g_k = f_k$, can be realized as

$$f_k = \begin{cases} \eta g_k + (1 - \eta) S_k & \text{if } \|g_k - S_k\| > \varepsilon_k \\ g_k & \text{otherwise.} \end{cases} \quad (14)$$

The BDCT domain quantization constraint set for the 8×8 image block of f can be expressed as [4]

$$C_Q = \{f : F_i^{\min} \leq (Df)_i \leq F_i^{\max}, \forall i = 1, 2, \dots, 64\} \quad (15)$$

where D is the BDCT transformation matrix, the subscript i is the frequency index, F_i^{\min} and F_i^{\max} are the i th minimum and maximum allowable BDCT coefficients as determined by the quantizer's i th quantization interval Δ_i and the transmitted BDCT symbol q_i , i.e.,

$$\begin{aligned} F_i^{\min} &= (q_i - 0.5)\Delta_i \\ F_i^{\max} &= (q_i + 0.5)\Delta_i \end{aligned} \quad (16)$$

where the transmitted symbol q_i is obtained by the quantization operation $q_i = \text{Round}[F_i^o / \Delta_i]$ on the i th BDCT coefficient, i.e., F_i^o , of the original image. Note that it is the set of symbols q that is transmitted to the decoder. Let $F = Df$, the projection of f onto the set C_Q is given by

$$P_Q f = D^T \hat{F} \quad (17)$$

where

$$\hat{F}_i = \begin{cases} F_i^{\min}, & \text{if } F_i < F_i^{\min} \\ F_i^{\max}, & \text{if } F_i > F_i^{\max} \\ F_i, & \text{otherwise.} \end{cases} \quad (18)$$

As the intensity of the original image is assumed to lie between 0 and 255, a range constraint set can be defined as $C_R = \{f : 0 \leq f_j \leq 255, j = 1, \dots, n_1 n_2\}$, where $n_1 n_2$ is the dimension of the image. The projection onto C_R , i.e., $P_R g = f$, can be realized as

$$f_j = \begin{cases} 0 & \text{if } g_j < 0 \\ 255 & \text{if } g_j > 255 \\ g_j & \text{otherwise.} \end{cases} \quad (19)$$

The deblocked image is obtained by iterating $f_n = P_R P_Q P_S f_{n-1}$, with f_0 given by the thin-plate spline image S , until convergence is reached.

The use of the thin-plate spline image S as the starting point f_0 instead of the BDCT coded image allows the proposed POCS algorithm to converge within a single iteration. We have the following proposition:

Proposition: Ignoring the projection P_R , and let $T = P_Q P_S$, the iteration $f_n = T f_{n-1}$, with $f_0 = S$, converges within a single iteration. Moreover, the error bound ε_k for the set C_s^k in (14) is immaterial.

Proof: The proof is by geometric argument. C_s^k is a hyper-sphere in a 64-dimension space, with center S_k and radius ε_k . Let f be an element on the convex surface of C_Q closest to C_s^k . Then $f = P_Q S_k$. Consider the path l joining S_k and f . Let t be the point where l intersects the surface of the hyper-sphere. Since l emanates from the center of the hyper-sphere, it is orthogonal to the tangential hyper-plane at t , and t is the closest point to f . Hence, the projection from f to the hyper-sphere C_s^k is also along the path l and landed on the point t , and the projection to $C_Q, P_Q t$, will bring us back to f again. Thus, the iteration converges within a single projection T using the initial point $f_0^k = S_k$. It is obvious that the radius ε_k is

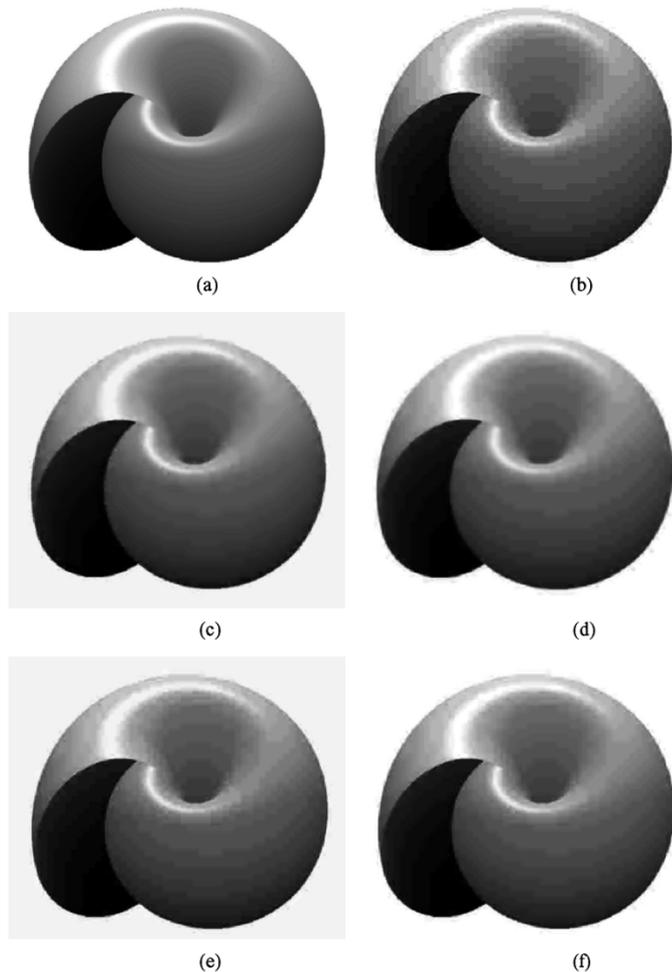


Fig. 2. Deblocked results for BDCT coded “SHELL” image. (a) Original 512×512 image. (b) BDCT coded image using Q2. Deblocked images from: (c) the proposed algorithm, (d) Zakhor’s algorithm, (e) Yang’s algorithm, and (f) MPEG4-MV algorithm.

immaterial. Since T is applied to each of the 8×8 image blocks independently, Proposition 1 extends to the entire image. \square

Although Proposition 1 ignores the set C_R , the conclusion still hold in practice when C_R is taken into account. This is due to the fact that the projected point $f \in C_Q$ is already “almost in” C_R , such that $P_R f \approx f$ in practice.

Intuitively, since the region smoothness constraint is applied to the individual image block, different error bound ε_k can be used for different block, depending on the degree of smoothness of the block. This consideration will speed up the convergence of the algorithm when the initial starting point is other than S , say, $f_0 = \text{BDCT coded image}$. However, proposition 1 shows that the value of ε_k can be disregard when the iteration starts with $f_0 = S$.

The theory of POCS assumes that the constraint sets intersect each other. However, intersection of the constraint sets is usually not guaranteed when they are defined. This is not a problem in practice, however, since $\|f_n - f_{n-1}\| \rightarrow 0$ when $n \rightarrow \infty$ even though the solution oscillates between each individual projection when the constraint sets do not intersect.

A final observation is that, if we shift the center of C_s^k to the projected point t on the surface of C_s^k after each iteration, then the constraint sets C_s^k and C_Q will eventually intersect each

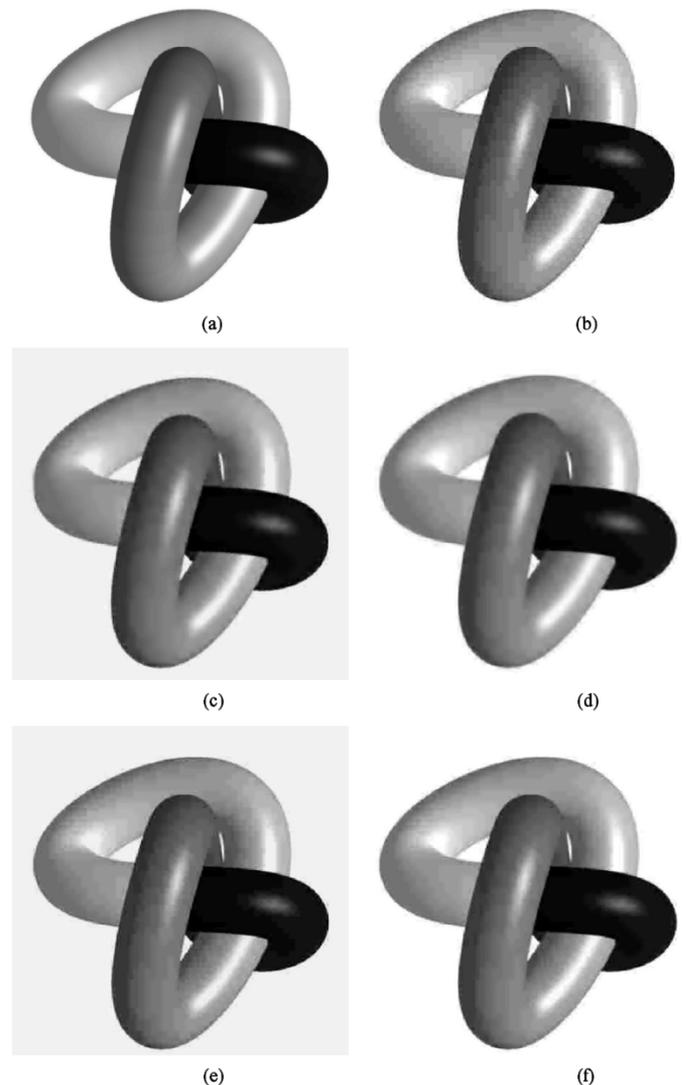


Fig. 3. Deblocked results for BDCT coded “KNOT” image. (a) Original 512×512 image. (b) BDCT coded image using Q2. Deblocked images from: (c) the proposed algorithm, (d) Zakhor’s algorithm, (e) Yang’s algorithm, and (f) MPEG4-MV algorithm.

other after a number of iterations. Note, however, that the projected point on C_Q (and hence the final solution) still stays fixed after the first iteration. This behavior can be easily visualized geometrically.

IV. EXPERIMENTAL RESULTS

The proposed algorithm has been tested on different BDCT coded graphic images containing objects with smooth regions. For illustration purpose, we present the deblocking results of two graphic images, “SHELL” and “KNOT”. The objects in the two graphic images are made up of smooth surfaces. The performance of the proposed algorithm is compared with several deblocking algorithms reported in the literature, namely: 1) Zakhor’s iterative deblocking algorithm [4]; 2) Yang’s spatially adaptive POCS algorithm [6]; and 3) the MPEG4-VM postfiltering algorithm [9]. Figs. 2 and 3 show the original image, the BDCT coded image using Q2, and the deblocked images for “SHELL” and “KNOT,” respectively. It can be seen that Zakhor’s algorithm generally produces an

TABLE I
PSNR PERFORMANCE (IN DECIBELS) OF DIFFERENT DEBLOCKING ALGORITHMS IN COMPARISON WITH OUR ALGORITHM. ALTHOUGH PSNR IS NOT INDICATIVE OF THE VISUAL QUALITY, OUR ALGORITHM NEVERTHELESS COMPARED FAVORABLY

Quantizer	SHELL			KNOT		
	Q1	Q2	Q3	Q1	Q2	Q3
BDCT coded	36.90	34.94	30.82	35.88	34.22	30.09
Zakhor [4]	35.26	33.83	29.72	34.49	33.18	29.18
Yang [6]	38.62	37.50	34.90	37.30	36.35	33.52
MPEG4 [9]	37.94	35.72	31.40	36.83	34.90	30.70
Our Method	38.58	37.51	34.89	37.27	36.33	33.53

TABLE II
MSDS ($\times 10^6$) PERFORMANCE OF DIFFERENT DEBLOCKING ALGORITHMS IN COMPARISON WITH OUR ALGORITHM. THE ORIGINAL MSDS FOR "SHELL" AND "KNOT" ARE 1.3225×10^6 AND 1.0541×10^6 , RESPECTIVELY

Quantizer	SHELL			KNOT		
	Q1	Q2	Q3	Q1	Q2	Q3
BDCT coded	5.4539	6.0028	10.651	6.5244	6.9600	12.888
Zakhor [4]	1.0542	1.1222	0.7693	1.1438	1.1183	0.82099
Yang [6]	1.5024	1.6242	1.9981	2.8108	2.9121	2.9785
MPEG4 [9]	3.7291	3.8329	5.6079	4.3092	4.3529	6.9110
Our Method	1.1619	1.1821	1.4881	1.4547	1.3943	1.9554

TABLE III
QUANTIZATION TABLES FOR BDCT

Q1								Q2								Q3							
50	60	70	70	90	120	255	255	86	59	54	86	129	216	255	255	110	130	150	192	255	255	255	255
60	60	70	96	130	255	255	255	64	64	75	102	140	255	255	255	130	150	192	255	255	255	255	255
70	70	80	120	200	255	255	255	75	70	86	129	216	255	255	255	150	192	255	255	255	255	255	255
70	96	120	145	255	255	255	255	75	91	118	156	255	255	255	255	192	255	255	255	255	255	255	255
90	130	200	255	255	255	255	255	97	118	199	255	255	255	255	255	255	255	255	255	255	255	255	255
120	255	255	255	255	255	255	255	129	189	255	255	255	255	255	255	255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255

overly blurred deblocked image due to the lowpass filtering operation, whereas the deblocked image obtained by Yang's algorithm and MPEG4-MV algorithm still contain significant blockiness in the smooth region. In contrast, our algorithm significantly reduces blockiness in the smooth region while still preserving the sharpness of the edges. It is worth pointing out that our algorithm is even better than Zakhor's algorithm (which tends to over-smooth) in suppressing blockiness in the smooth regions in both test images.

For performance evaluation, we used PSNR and MSDS introduced in [18]. Unlike PSNR which involves pixels in the entire image, the MSDS involves just pixels near the block boundaries and has been proposed to be a better measure of the impact of blocking effects [18]–[20]. Consider an 8×8 block z . Let n, s, w, e be the blocks adjacent to z , in the directions north, south, west and east, respectively. Then, MSDS is defined as [18],

$$\text{MSDS} = \varepsilon_w + \varepsilon_e + \varepsilon_n + \varepsilon_s \quad (20)$$

where

$$\varepsilon_w = \sum_{m=0}^7 [d_1(m) - d_2(m)]^2 \quad (21)$$

$$d_1(m) = z(m, 0) - w(m, 7) \quad (22)$$

$$d_2(m) = 0.5 * (w(m, 7) - w(m, 6) + z(m, 1) - z(m, 0)) \quad (23)$$

and $\varepsilon_e, \varepsilon_n$, and ε_s are defined similarly.

Tables I and II show PSNR and MSDS comparison using quantization tables in the Appendix, respectively. It can be observed that the proposed algorithm compared favorably to the other algorithms in term of PSNR and MSDS. For the PSNR measure, our algorithm ranks within the top two performers. For the MSDS measure, we see that Zakhor's algorithm generally over smooth the image, while MPEG4-MV algorithm under smooth the image. Yang's algorithm performs fairly well but the reconstructed image is still not smooth enough compared to the result from our algorithm.

In terms of computation complexity, the MPEG4 algorithm is the fastest since it is noniterative and it works directly in the image domain. The other two algorithms (Yang and Zakhor) are iterative POCS-based methods similar to our algorithm. Although our algorithm has higher computation cost due to the explicit modeling of the smooth surface using thin plate spline, Yang and Zakhor's algorithms in general require between six to ten iterations to converge to a stable solution.

V. CONCLUSION

In this paper, we proposed a novel POCS-based deblocking algorithm for graphic images with smooth surfaces. Our algorithm employs a smoothness constraint that involves explicit modeling of smooth regions in the image using a spatially adaptive thin-plate spline. By adapting the sampling point

configuration and the regularization weight to the degree of smoothness of the underlying image blocks, the tps can be made adaptive to the smoothness of the underlying image. As the smoothness constraint is enforced over a region instead of just around the 8×8 block boundaries, better blockiness suppression is achieved. We showed that the convergence of our algorithm could be reached within one iteration. Visual and quantitative evaluations (in terms of PSNR and MSDS) indicate that the algorithm can effectively suppress blockiness in smooth regions while still preserve the sharpness of edges.

APPENDIX QUANTIZATION TABLES FOR BDCT

See Table III.

REFERENCES

- [1] N. Ahmad, T. Natarajan, and K. R. Rao, "Discrete cosine transform coding of images," *IEEE Trans. Comput.*, vol. C-23, pp. 90-93, 1941.
- [2] H. C. Reeves and J. S. Lim, "Reduction of blocking effects in image coding," *J. Opt. Eng.*, vol. 23, pp. 34-37, Jan. 1984.
- [3] N. F. Law and W. C. Siu, "Successive structural analysis using wavelet transform for blocking artifacts suppression," *Signal Process.*, vol. 81, no. 7, pp. 1373-1387, Jul. 2001.
- [4] A. Zakhor, "Iterative procedures for reduction of blocking effects in transform image coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 2, no. 3, pp. 91-95, Mar. 1992.
- [5] Y. Yang, N. P. Galatsanos, and A. K. Katsaggelos, "Regularized reconstruction to reduce blocking artifacts of block discrete cosine transform compressed images," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 3, no. 6, pp. 431-442, Dec. 1993.
- [6] Y. Yang and N. P. Galatsanos, "Projection-based spatially adaptive reconstruction of block-transform compressed images," *IEEE Trans. Image Process.*, vol. 4, no. 7, pp. 896-908, Jul. 1995.
- [7] S. D. Kim, J. Yi, H. M. Kim, and J. B. Ra, "A deblocking filter with two separate mode in block-based video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 2, pp. 156-160, Feb. 1999.
- [8] H. W. Park and Y. L. Lee, "A postprocessing method for reducing quantization effects in low bit-rate moving picture coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 2, pp. 161-171, Feb. 1999.
- [9] MPEG 4 Verification Model, VM 14.2, ISO/IEC JTC1/SC29/WG11 N3908. (1999). [Online]. Available: http://www.chiariglione.org/mpeg/working_documents.htm
- [10] A. W. C. Liew and H. Yan, "Blocking artifacts suppression in block-coded images using overcomplete wavelet representation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 4, pp. 450-461, Apr. 2004.
- [11] C. Weerasinghe, A. W. C. Liew, and H. Yan, "Artifact reduction in compressed images based on region homogeneity constraints using the projections onto convex sets algorithm," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. , pp. 891-897, Oct. 2002.
- [12] F. L. Bookstein, "Principal warps: Thin-plate splines and the decomposition of deformations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 11, no. 6, pp. 567-585, Jun. 1989.
- [13] J. Meinguet, "Surface spline interpolation: Basic theory and computational aspects," in *Approximation Theory and Spline Functions*, S. P. Singh, Ed. Dordrecht, The Netherlands: Reidel, 1984, pp. 127-142.
- [14] A. W. C. Liew and N. F. Law, "Reconstruction from 2-D wavelet transform modulus maxima using projection," in *Proc. IEE Vision, Image, and Signal Processing*, vol. 147, Apr. 2000, pp. 176-184.
- [15] A. Liew and D. T. Nguyen, "Reconstruction from wavelet transform modulus maxima using nonexpansive projections," *Electron. Lett.*, vol. 31, no. 13, pp. 1038-1039, Jun. 22, 1995.
- [16] H. Stark and Y. Yang, *Vector Space Projections, a Numerical Approach to Signal and Image Processing, Neural Nets, and Optics*. New York: Wiley, 1998.
- [17] D. C. Youla and H. Webb, "Image restoration by the method of convex projections: Part I theory," *IEEE Trans. Med. Imag.*, vol. 1, no. 2, pp. 81-94, Feb. 1982.
- [18] S. Minami and A. Zakhor, "An optimization approach for removing blocking effects in transform coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 5, no. 4, pp. 74-82, Apr. 1995.
- [19] G. Lakhani and N. Zhong, "Derivation of prediction equations for blocking effect reduction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 4, pp. 415-418, Apr. 1999.
- [20] G. A. Triantafyllidis, D. Tzovaras, and M. G. Strintzis, "Blocking artifact detection and reduction in compressed data," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 10, pp. 877-890, Oct. 2002.
- [21] F. Girosi, M. Jones, and T. Poggio, "Regularization theory and neural networks architectures," *Neural Comput.*, vol. 7, no. 2, pp. 219-269, Mar. 1995.
- [22] G. Wahba, *Spline Models for Observational Data*. Philadelphia, PA: SIAM, 1990.