

Graph-based Dequantization of Block-Compressed Piecewise Smooth Images

Wei Hu, *Student Member, IEEE*, Gene Cheung, *Senior Member, IEEE*, and Masato Kazui, *Member, IEEE*

Abstract—Block-based image or video coding standards (e.g. JPEG) compress an image lossily by quantizing transform coefficients of non-overlapping pixel blocks. If the chosen quantization parameters (QP) are large, then hard decoding of a compressed image—using indexed quantization bin centers as reconstructed transform coefficients—can lead to unpleasant blocking artifacts. Leveraging on recent advances in graph signal processing (GSP), we propose a dequantization scheme specifically for piecewise smooth (PWS) images: images with sharp object boundaries and smooth interior surfaces. We first mathematically define a PWS image as a low-frequency signal with respect to an inter-pixel similarity graph with edges of weights 1 or 0. Using quantization bin boundaries as constraints, we then jointly optimize the desired graph-signal and the similarity graph in a unified framework. A generalization to consider generalized piecewise smooth (GPWS) images—where sharp object boundaries are replaced by transition regions—is also proposed. Experimental results show that our proposed scheme outperforms a state-of-the-art dequantization method by 1 dB on average in PSNR.

Index Terms—Graph signal processing, image coding.

I. INTRODUCTION

BLOCK-BASED image or video compression standards like JPEG¹ first divide an image into non-overlapping pixel blocks, project each block onto basis functions of a chosen frequency domain such as discrete cosine transform (DCT), and quantize the resulting transform coefficients for compression gain. If the chosen quantization parameters (QP) are large, then *hard decoding* of a compressed image at the decoder—using indexed quantization bin centers as reconstructed transform coefficients—can result in unpleasant blocking artifacts.

In contrast, *soft decoding* approaches [1]–[5] treat image dequantization as an under-determined inverse problem: find the most probable transform coefficients in a code block subject to indexed quantization bin constraints, given suitably defined signal priors. [1] assumes band-limitedness in the targeted

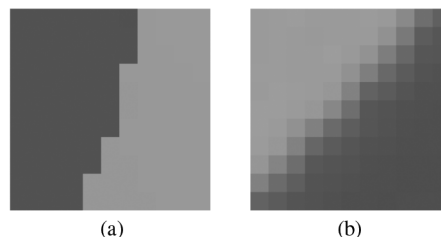


Fig. 1. Examples of piecewise smooth (PWS) and generalized piecewise smooth (GPWS) pixel patches. (a) PWS patch, (b) GPWS patch.

code block and performs projection on convex sets (POCS) between quantization bin constraints and a band-limited subspace. The band-limited assumption is too strong and results in over-smoothing, however. [2] assumes a total variation (TV) prior, which performs well only for piecewise constant functions. [3]–[5] assume a sparse signal model, where the targeted block is approximated as a sparse combination of atoms from an over-complete dictionary. Computing a sparse code vector from a large unstructured dictionary, however, is computationally intensive. There are also nonlocal methods [6] that exploit self-similarity of pixel patches across different spatial regions to remove blocking artifacts in a hard-decoded image. The search for similar nonlocal patches is expensive in computation and buffer space, however.

Leveraging on recent advances in graph signal processing (GSP) [7], we propose a *graph-signal smoothness prior* for soft decoding of JPEG compressed images; retrofitting our algorithm to dequantize other block-compressed images is straightforward. In particular, we focus on *piecewise smooth* (PWS) images, where smooth textural regions are separated by sharp discontinuities. See Fig. 1(a) for an example. PWS images include depth images, graphics images and sub-regions of video frames overlaid with foreign language captions. Structure of PWS pixel patches can be modeled using simple graphical models—4-connected graphs with edge weights either 0 or 1 that reflect inter-pixel similarities [8]. Assuming that the desired signal contains mostly low frequencies with respect to a graph, we propose a unified framework that alternately optimizes the graph (image structure) and the signal on top of the graph (pixel patch) while satisfying the quantization bin constraints. In particular, we show that the signal can be optimized efficiently via quadratic programming, while the graph can be optimized via a fast max-flow / min-cut algorithm [9] in polynomial time. Moreover, unlike nonlocal methods [6] our optimization is performed locally, thus amenable to buffer-constrained hardware implementation.

Compared to recent graph-based denoising [10]–[13] and dequantization works [4], we differ in two major ways. First, unlike [4] that learns an over-complete dictionary for a sparse signal prior that depends on training patches in natural images,

Manuscript received October 12, 2015; revised December 04, 2015; accepted December 05, 2015. Date of publication December 23, 2015; date of current version January 07, 2016. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Yao Zhao.

W. Hu is with Department of Electronic and Computer Engineering, The Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong, China (e-mail: huwei@ust.hk).

G. Cheung is with National Institute of Informatics, Tokyo 101–8430, Japan (e-mail: cheung@nii.ac.jp).

M. Kazui is with Samsung R&D Institute Japan, Yokohama 230–0027, Japan (e-mail: m.kazui@samsung.com).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/LSP.2015.2510379

¹<http://www.jpeg.org/jpeg/>

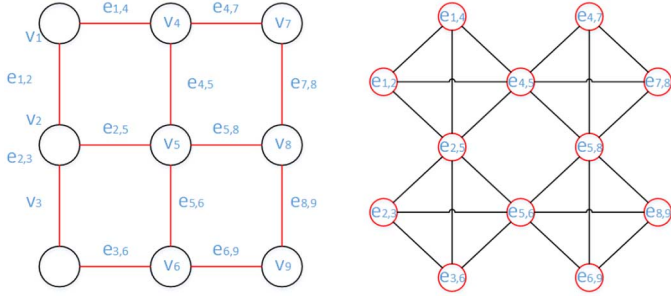


Fig. 2. (Left) 4-connectivity graph \mathcal{G} for 3×3 block. (Right) Dual graph \mathcal{G}_w .

we design an edge weight prior specifically for PWS images, which does not require offline dictionary training and online sparse vector search, lowering overall complexity. Second, we jointly optimize *both* the graph-signal *and* the graph that describes the signal kernel, while [4], [10]–[13] optimize the signal assuming first that the appropriate graph can be deduced from noisy observations or updated version of the signal. *To the best of our knowledge, we are the first in the graph-based inverse imaging literature to explicitly include both the graph and the graph-signal in the optimization objective.*

We next generalize our formulation to consider also *generalized piecewise smooth* (GPWS) images, where sharp object boundaries are replaced by a transition region. See Fig. 1(b) for an illustration. GPWS images include cartoon images, medical images like X-ray and MRI, and saliency maps. We show that a similar optimization algorithm can be used for GPWS images. Experimental results show that our proposed scheme outperforms a state-of-the-art dequantization method [6] by 1 dB on average in PSNR.

II. PROBLEM FORMULATION FOR PWS IMAGES

We first define the prior probability for edge weights in a similarity graph for a PWS pixel patch. We then define a smoothness prior for graph-signals with respect to a given graph. We formulate a *maximum a posteriori* (MAP) estimation problem for the desired PWS pixel patch. Finally, we describe a fast algorithm to solve the formulated problem.

A. Graph Construction for PWS Pixel Patch

A graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathbf{W}\}$ is composed of a vertex set \mathcal{V} , an edge set \mathcal{E} connecting vertices, and a weighted adjacency matrix \mathbf{W} , where $w_{i,j}$ is the weight assigned to the edge connecting vertices i and j . We build \mathcal{G} on a $\sqrt{N} \times \sqrt{N}$ PWS pixel patch, $\sqrt{N} \in \mathbb{Z}^+$, as follows. We represent each pixel z_i , $1 \leq i \leq N$, as a vertex i in \mathcal{G} , and connect two vertices i and j with an edge $e_{i,j}$ in \mathcal{E} iff the corresponding pixels z_i and z_j are adjacent neighbors in the horizontal or vertical direction on the 2D grid. This results in a 4-connectivity graph as shown in Fig. 2 (left). Each edge weight reflects the similarity between the two connecting pixels (e.g., setting $w_{i,j} = \exp\{-\|z_i - z_j\|_2^2 / \sigma^2\}$ [10], [11], [14] yields $w_{i,j} \approx 1$ if the connecting pixels i and j have similar intensities z_i and z_j); we call \mathcal{G} the *similarity graph*. Specifically for PWS images, adjacent pixels in smooth regions are similar and we set the corresponding edge weights $w_{i,j} = 1$, and adjacent pixels across sharp boundaries are dissimilar and we set $w_{i,j} = 0$.

We now define the prior probability of edge weights of a similarity graph for a PWS pixel patch. We stack all K edge

weights² $w_{i,j}$ in \mathcal{G} into a vector $\mathbf{w} \in \mathbb{R}^K$. We treat \mathbf{w} as another graph-signal, and define it on a *dual graph* \mathcal{G}_w , where edges in \mathcal{G} become *nodes*³ in \mathcal{G}_w , as shown in Fig. 2 (right). We then draw *links* to connect nodes in \mathcal{G}_w that represent neighboring edges in \mathcal{G} . Specifically, we draw a link of weight 1 between two nodes in \mathcal{G}_w iff the two corresponding edges in \mathcal{G} share one same vertex as an endpoint, *or* two endpoints of the corresponding edge of one node are *both* one-hop neighbors of the endpoints of the corresponding edge of the other node.

On \mathcal{G}_w , we define priors for both the AC and DC⁴ components of \mathbf{w} to characterize piecewise smoothness. First, boundaries of objects in a PWS image are typically contiguous, which means that neighboring edge weights in original \mathcal{G} (nodes in dual \mathcal{G}_w) are likely similar. We thus define the AC prior as the *total variation* (TV) of \mathbf{w} :

$$f(\mathbf{w}_A) = c_A \exp\{-\mu_2 \|\mathbf{H}\mathbf{w}\|_1\}, \quad (1)$$

where \mathbf{H} is a $S \times K$ difference matrix— S being the number of links in \mathcal{G}_w —that computes the difference in values between each pair of connected nodes in \mathcal{G}_w . c_A is a normalization factor and μ_2 is a parameter.

Second, since most regions are smooth in PWS images, most edge weights are 1. Hence, we define the DC prior as

$$f(\mathbf{w}_D) = c_D \exp\{-\mu_3 \|\mathbf{1} - \mathbf{w}\|_1\}, \quad (2)$$

where $\mathbf{1} \in \mathbb{R}^K$ is an all-one vector, c_D is a normalization factor and μ_3 is a parameter.

The prior for \mathbf{w} is the product of AC and DC priors:

$$f(\mathbf{w}) = c_2 \exp\{-\mu_2 \|\mathbf{H}\mathbf{w}\|_1\} \exp\{-\mu_3 \|\mathbf{1} - \mathbf{w}\|_1\}, \quad (3)$$

where c_2 is a normalization factor, and we constrain the feasible space of each element w_i to be $\{0, 1\}$ (0 for sharp discontinuities, and 1 for smooth regions).

B. Graph-Signal Smoothness Prior for PWS Images

We declare a pixel patch \mathbf{z} is PWS iff \mathbf{z} is smooth with respect to a similarity graph \mathcal{G} with edge weights \mathbf{w} of probability $f(\mathbf{w})$ in (3). Mathematically, \mathbf{z} is smooth iff

$$\sum_{i \sim j} w_{i,j} (z_i - z_j)^2 < \epsilon, \forall i, j \quad (4)$$

where ϵ is a threshold of a small positive value, and $i \sim j$ means two vertices i and j are one-hop neighbors in \mathcal{G} . In order to satisfy (4), z_i and z_j must be similar for a large $w_{i,j}$, but could be dissimilar for a small $w_{i,j}$.

We define the combinatorial graph Laplacian as $\mathcal{L} := \mathbf{D} - \mathbf{W}$ [15], where \mathbf{D} is the *degree matrix*—a diagonal matrix where $d_{i,i} = \sum_{j=1}^N w_{i,j}$. As $\mathbf{z}^T \mathcal{L}(\mathbf{w}) \mathbf{z} = \sum_{i \sim j} w_{i,j} (z_i - z_j)^2$ [16], (4) can be concisely written as $\mathbf{z}^T \mathcal{L}(\mathbf{w}) \mathbf{z} < \epsilon$.

The prior distribution for \mathbf{z} given \mathbf{w} can now be written as

$$f(\mathbf{z}|\mathbf{w}) = c_1 \exp\{-\mu_1 \mathbf{z}^T \mathcal{L}(\mathbf{w}) \mathbf{z}\}, \quad (5)$$

where c_1 is a normalization factor and μ_1 is a parameter. We refer to (5) as the *conditional graph-signal smoothness prior* of

³As a convention, we use terminologies “vertices” and “edges” for the original graph \mathcal{G} , and “nodes” and “links” for the dual graph \mathcal{G}_w .

⁴“DC” (direct current) means the 0-frequency component of a signal \mathbf{w} , and “AC” (alternating current) means the higher-frequency components.

²For a $\sqrt{N} \times \sqrt{N}$ 4-connectivity graph, there are $K = 2(N - \sqrt{N})$ edges.

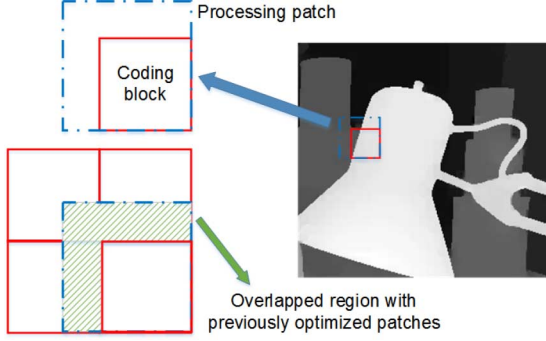


Fig. 3. A patch encloses a 8×8 coding block in JPEG. The current patch overlaps with previously optimized patches, and consistency of the overlapped regions among these patches is enforced during optimization.

\mathbf{z} . Combining (3) and (5), we write the *graph-signal smoothness prior* $f(\mathbf{z})$ using the law of total probability:

$$f(\mathbf{z}) = \sum_{\mathbf{w}} f(\mathbf{z}|\mathbf{w})f(\mathbf{w}) \quad (6)$$

C. MAP Estimation of PWS Images

As shown in Fig. 3, we operate on a $\sqrt{N} \times \sqrt{N}$ *processing unit* \mathbf{z} —a patch that encompasses a smaller $\sqrt{M} \times \sqrt{M}$ ($M < N$) coding block \mathbf{y} (8×8 block in JPEG) that was transformed by the DCT matrix \mathbf{T} and quantized at the encoder. This is done to remove blocking artifacts (inconsistent reconstructed signals across block boundaries), as done in [4]. The DCT coefficients of \mathbf{y} is written as $\boldsymbol{\alpha} = \mathbf{T}\mathbf{E}\mathbf{z}$, where \mathbf{E} is a $M \times N$ matrix that extracts \mathbf{y} from \mathbf{z} .

At the decoder, the received quantization bin centers β and quantization parameters \mathbf{q} together define the indexed quantization bins $\mathbb{F}(\beta, \mathbf{q}) = [\beta - \frac{\mathbf{q}}{2}, \beta + \frac{\mathbf{q}}{2})$ for coefficients $\boldsymbol{\alpha}$. We then pose a MAP estimation problem: given indexed quantization bins $\mathbb{F}(\beta, \mathbf{q})$, find the most probable signal \mathbf{z} ,

$$\hat{\mathbf{z}}_{\text{MAP}}(\beta, \mathbf{q}) = \arg \max_{\mathbf{z}} f(\beta, \mathbf{q}|\mathbf{z})f(\mathbf{z}) \quad (7)$$

where $f(\beta, \mathbf{q}|\mathbf{z})$ is the likelihood term, and $f(\mathbf{z})$ is the signal prior (6) derived previously. In dequantization, likelihood takes a simple form; it is 1 iff the block coefficients $\boldsymbol{\alpha}$ are inside the indexed quantization bins, and 0 otherwise:

$$f(\beta, \mathbf{q}|\mathbf{z}) = \begin{cases} 1, & \text{if } \mathbf{T}\mathbf{E}\mathbf{z} \in \mathbb{F}(\beta, \mathbf{q}) \\ 0, & \text{otherwise} \end{cases} \quad (8)$$

We can thus rewrite (7) as a sum of exponentials, where:

$$\begin{aligned} w_i &\in \{0, 1\}, \forall i: \\ &= \max_{\mathbf{T}\mathbf{E}\mathbf{z} \in \mathbb{F}(\beta, \mathbf{q})} f(\mathbf{z}) = \max_{\mathbf{T}\mathbf{E}\mathbf{z} \in \mathbb{F}(\beta, \mathbf{q})} \sum_{\mathbf{w}} f(\mathbf{z}|\mathbf{w})f(\mathbf{w}) \\ &= \max_{\mathbf{T}\mathbf{E}\mathbf{z} \in \mathbb{F}(\beta, \mathbf{q})} \sum_{\mathbf{w}} \exp\{-\mu_1 \mathbf{z}^T \mathcal{L}(\mathbf{w})\mathbf{z} \\ &\quad - \mu_2 \|\mathbf{H}\mathbf{w}\|_1 - \mu_3 \|\mathbf{1} - \mathbf{w}\|_1\}, \end{aligned} \quad (9)$$

1) *Problem Formulation*: Solving (9) requires a summation over multiple weight vectors \mathbf{w} each with edge weight prior $f(\mathbf{w})$. Instead, we make the following approximation:

$$\max_{\mathbf{T}\mathbf{E}\mathbf{z} \in \mathbb{F}(\beta, \mathbf{q})} \sum_{\mathbf{w}} f(\mathbf{z}|\mathbf{w})f(\mathbf{w})$$

$$\approx \max_{\mathbf{T}\mathbf{E}\mathbf{z} \in \mathbb{F}(\beta, \mathbf{q})} \max_{\mathbf{w}} f(\mathbf{z}|\mathbf{w})f(\mathbf{w}) \quad (10)$$

This approximation is good if the distribution $f(\mathbf{z}|\mathbf{w})f(\mathbf{w})$ is concentrated around a single peak [14]. In our case, $f(\mathbf{z}|\mathbf{w})$ decays quickly from the peak, as $\mathbf{z}^T \mathcal{L}(\mathbf{w})\mathbf{z}$ becomes large when the structure of \mathbf{z} is inconsistent with \mathbf{w} in (5). Thus the approximation is reasonable in our case.

Given the approximation in (10), we take the negative log—turning maximization to minimization—resulting in the following problem formulation:

$$\begin{aligned} \min_{\mathbf{z}, \mathbf{w}} \quad & \mu_1 \mathbf{z}^T \mathcal{L}(\mathbf{w})\mathbf{z} + \mu_2 \|\mathbf{H}\mathbf{w}\|_1 + \mu_3 \|\mathbf{1} - \mathbf{w}\|_1 \\ \text{s.t.} \quad & w_i \in \{0, 1\}, \forall i \\ & \beta - \frac{\mathbf{q}}{2} \leq \mathbf{T}\mathbf{E}\mathbf{z} \leq \beta + \frac{\mathbf{q}}{2}, \sum_{\mathbf{z}_k \in \mathcal{N}(\mathbf{z})} \|\mathbf{S}_{kz}\mathbf{z} - \mathbf{S}_{zk}\mathbf{z}_k\|^2 < \tau \end{aligned} \quad (11)$$

The last constraint is added to enforce local consistency in overlapped regions between \mathbf{z} and its neighboring patches $\mathbf{z}_k \in \mathcal{N}(\mathbf{z})$ optimized in previous iterations, as shown in Fig. 3. \mathbf{S}_{kz} is a matrix that extracts the region of \mathbf{z} overlapped with \mathbf{z}_k , and \mathbf{S}_{zk} extracts the region of \mathbf{z}_k overlapped with \mathbf{z} . τ is a small threshold.

D. Alternating Minimizing Algorithm

In order to solve (11) with two variables \mathbf{z} and \mathbf{w} representing the signal and the graph respectively, we propose an alternating minimizing algorithm to optimize one variable at a time while the other is fixed. We first initialize \mathbf{w} using spectral clustering [17]. Specifically, we divide a pre-processed \mathbf{z} into two segments via spectral clustering, and then assign weight 1 to edges in each segment, and assign weight 0 to edges across the two segments. Note that though the optimization is carried out on a $\sqrt{N} \times \sqrt{N}$ patch \mathbf{z} , we perform spectral clustering on a larger patch that encloses \mathbf{z} , as better clustering results can be achieved on a larger patch of larger variance.

We substitute the initialized \mathbf{w} into (11). Then the objective with respect to \mathbf{z} is quadratic, and there remain one linear constraint and one quadratic constraint—this is a quadratic programming problem that can be solved efficiently. Next, we fix the optimized \mathbf{z} in (11) and update \mathbf{w} . The resulting objective is convex with respect to \mathbf{w} and only the first discrete constraint remains—this is a standard *separation-deviation* (SD) problem and can be solved optimally using an efficient algorithm in [9] in polynomial time. \mathbf{z} and \mathbf{w} are optimized alternately until \mathbf{z} converges.

III. EXTENSION TO GENERALIZED PWS IMAGES

We extend the above problem formulation for PWS images to GPWS images. As shown in Fig. 1, GPWS images differ from PWS images in regions that separate two smooth regions: instead of ideal *sharp discontinuities* in PWS images, GPWS images contain *gradual transitions* from one smooth region to another. We thus change the AC prior probability of \mathbf{w} in \mathcal{G}_w accordingly: in the transition region weights may deviate from 1, and neighboring weights are similar in a l_1 -norm sense. Neighboring weights are connected nodes in \mathcal{G}_w as in Fig. 2(b). We then define $f_g(\mathbf{w}_A)$ for GPWS images as

$$f_g(\mathbf{w}_A) = \max_d f_d(\mathbf{w}) = \max_d c_d \exp\{-\lambda_d \|\mathbf{H}_d \mathbf{w}\|_1\}, \quad (12)$$

TABLE I
PERFORMANCE COMPARISON IN PSNR (dB) AT $QF = 15$

Images	$QF = 15$					
	JPEG	TV	DicTV	TGV	ANCE	Prop
Dude	37.00	34.23	37.58	37.19	37.99	38.17
Teddy	31.46	32.30	31.60	31.52	32.14	32.33
Tsukuba	33.13	35.29	34.19	33.68	34.69	36.22
Ballet	35.63	36.48	36.77	36.15	37.28	37.49
Champagne	36.82	34.12	37.46	37.00	37.73	37.68
Gain	1.57	1.89	0.86	1.27	0.41	-

TABLE II
PERFORMANCE COMPARISON IN PSNR (dB) AT $QF = 35$

Images	$QF = 35$					
	JPEG	TV	DicTV	TGV	ANCE	Prop
Dude	40.57	42.77	40.49	41.81	42.16	44.80
Teddy	34.15	36.81	34.00	36.49	34.80	36.91
Tsukuba	36.17	39.97	37.68	39.44	38.36	40.72
Ballet	39.47	42.04	40.82	39.94	41.12	41.76
Champagne	41.29	41.99	42.79	41.88	42.73	43.00
Gain	3.11	0.72	2.46	1.53	1.60	-

where d denotes nodes in a connected component in \mathcal{G}_w to specify a transition region in a GPWS patch. Nodes not in d are in smooth regions with weight 1. \mathbf{H}_d computes the difference between neighboring weights in d . c_d is a normalization factor and λ_d is a parameter. We denote $\hat{d} = \arg \max_d f_d(\mathbf{w})$, and formulate the dequantization problem for a GPWS patch as:

$$\begin{aligned}
 \min_{\mathbf{z}, \mathbf{w}} \quad & \mathbf{z}^T \mathcal{L}(\mathbf{w}) \mathbf{z} + \lambda_{\hat{d}} \|\mathbf{H}_{\hat{d}} \mathbf{w}\|_1 + \lambda_2 \|\mathbf{1} - \mathbf{w}\|_1 \\
 \text{s.t.} \quad & 0 \leq w_i \leq 1, \forall w_i \in \hat{d}, w_i = 1 \forall w_i \notin \hat{d} \\
 & \beta - \frac{\mathbf{q}}{2} \leq \mathbf{T} \mathbf{E} \mathbf{z} \leq \beta + \frac{\mathbf{q}}{2}, \sum_{\mathbf{z}_k \in \mathcal{N}(\mathbf{z})} \|\mathbf{S}_{kz} \mathbf{z} - \mathbf{S}_{zk} \mathbf{z}_k\|^2 < \tau
 \end{aligned} \tag{13}$$

We optimize one variable at a time. When initializing \mathbf{w} , we divide \mathbf{z} into *three* segments via spectral clustering. The third segment is the most probable transition region \hat{d} . Then edge weights in \hat{d} are not assigned 1 but exponential functions of the corresponding inter-pixel intensity differences, while edge weights in smooth segments are assigned 1. When \mathbf{w} and \hat{d} are fixed, \mathbf{z} is optimized via quadratic programming. When \mathbf{z} and \hat{d} are fixed, \mathbf{w} is optimized via convex programming. \hat{d} is then re-computed via spectral clustering for this patch only. The procedure repeats until convergence.

IV. EXPERIMENTATION

We first test our algorithm on five depth images: *Dude* of size 480×800 , *Teddy* of size 368×448 , *Sawtooth* of size 380×434 , *Tsukuba* of size 480×640 and *Champagne* of size 960×1280 . The parameters are empirically assigned as $\mu_1 = 1$, $\mu_2 = 1$, $\mu_3 = 5$, $\lambda_d = 1$ and $\lambda_2 = 5$. We compare against five methods: 1) JPEG hard decoding; 2) TV [2], the TV-based JPEG decomposition method; 3) DicTV [18], a recent sparsity-based restoration algorithm for decomposition, combining the TV prior with sparsity; 4) TGV [19], JPEG decomposition with Total Generalized Variation (TGV); and 5) ANCE algorithm [6], which is a state-of-the-art algorithm for compression artifact reduction.

Table I and II list PSNR results of these approaches using test images coded by a JPEG encoder with quality factor (QF) 15 and 35, respectively. Larger QF value means smaller quanti-

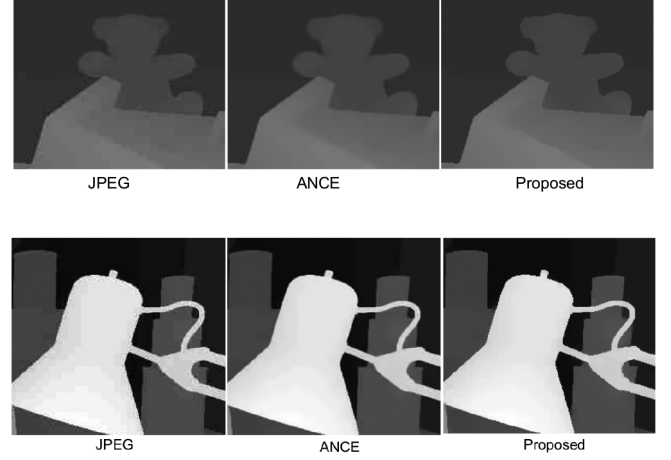


Fig. 4. Visual comparison among different dequantization methods. First row: *Teddy* at $QF = 25$; Second row: *Tsukuba* at $QF = 15$.

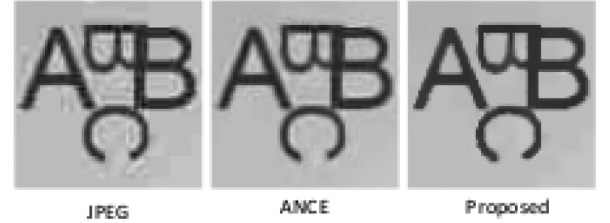


Fig. 5. The subjective quality comparison among different dequantization methods for a GPWS image *ColoText* at $QF = 15$.

TABLE III
PERFORMANCE COMPARISON IN PSNR (dB) AT $QF = 15$

Images	$QF = 15$						
	JPEG	TV	DicTV	TGV	ANCE	PWS	Prop
ColoText	29.05	32.22	30.66	31.03	30.91	29.15	32.81
Mickey	29.42	30.85	28.94	29.90	30.95	29.18	31.04
Kanji	24.67	26.92	24.40	25.13	24.87	24.45	27.07
Gain	2.59	0.31	1.54	1.62	1.40	2.71	-

zation bins and thus better image quality. The bold numbers are the best PSNR among competing schemes.

We observe that our method results in the best PSNR in most cases. On average, we achieve 2.36 dB gain over JPEG, 1.12 dB gain over TV, 1.62 dB gain over DicTV, 1.40 dB gain over TGV and 1.40 dB gain over ANCE. Note that JPEG, TV, TGV and our method are local methods while DicTV and ANCE are non-local methods. Though non-local methods generally lead to better results than local methods, our algorithm still outperforms one state-of-the-art nonlocal method ANCE.

Fig. 4 shows dequantized image regions. Unlike other methods, our results reconstruct sharp edges noticeably well. This is due to the effectiveness of our proposed graph-signal smoothness prior.

We also test our extended method on GPWS images: *ColoText* (printed text) of size 256×256 , *Mickey* (animation) of size 512×512 and *Kanji* (foreign language caption) of size 56×72 . Table III lists PSNR results of different approaches at $QF = 15$. On average we achieve 2.59 dB gain over JPEG, 0.31 dB gain over TV, 1.54 dB gain over DicTV, 1.40 dB gain over ANCE, and 2.71 dB gain over our proposed method PWS for PWS images. Fig. 5 shows our proposed method well preserves the transition regions in *ColoText*.

REFERENCES

- [1] A. Zakhori, "Iterative procedures for reduction of blocking effects in transform image coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 2, no. 1, pp. 91–95, Mar. 1992.
- [2] K. Bredies and M. Holler, "A total variation-based JPEG decompression model," *SIAM J. Img. Sci.*, vol. 5, no. 1, pp. 366–393, Mar. 2012.
- [3] G. Farinella and S. Battiato, "On the application of structured sparse model selection to JPEG compressed images," in *Computational Color Imaging*. Berlin/Heidelberg, Germany: Springer, 2011, vol. 6626 of Lecture Notes in Computer Science.
- [4] X. Liu, G. Cheung, X. Wu, and D. Zhao, "Inter-block soft decoding of JPEG images with sparsity and graph-signal smoothness priors," in *IEEE Int. Conf. Image Processing*, Quebec City, Canada, Sep. 2015.
- [5] X. Liu, X. Wu, J. Zhou, and D. Zhao, "Data-driven sparsity-based restoration of JPEG-compressed images in dual transform-pixel domain," in *IEEE Conf. Computer Vision and Pattern Recognition*, Boston, U.S., Jun. 2015.
- [6] X. Zhang, R. Xiong, X. Fan, S. Ma, and W. Gao, "Compression artifact reduction by overlapped-block transform coefficient estimation with block similarity," *IEEE Trans. Image Process.*, vol. 22, no. 12, pp. 4613–4626, Dec. 2013.
- [7] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs," *IEEE Signal Process. Mag.*, pp. 83–98, May 2013.
- [8] W. Hu, G. Cheung, A. Ortega, and O. Au, "Multi-resolution graph fourier transform for compression of piecewise smooth images," *IEEE Trans. Image Process.*, vol. 24, no. 1, pp. 419–433, Jan. 2015.
- [9] D. S. Hochbaum, "An efficient and effective tool for image segmentation, total variations and regularization," in *SSVM'11 Proc. Third Int. Conf. Scale Space and Variational Methods in Computer Vision*, 2011, pp. 338–349.
- [10] W. Hu, X. Li, G. Cheung, and O. Au, "Depth map denoising using graph-based transform and group sparsity," in *IEEE Int. Workshop on Multimedia Signal Processing*, Pula, Italy, Oct. 2013.
- [11] W. Hu, G. Cheung, X. Li, and O. C. Au, "Graph-based joint denoising and super-resolution of generalized piecewise smooth images," in *IEEE Int. Conf. Image Processing*, Paris, France, Oct. 2014.
- [12] J. Pang, G. Cheung, W. Hu, and O. C. Au, "Redefining self-similarity in natural images for denoising using graph signal gradient," in *APSIPA ASC*, Siem Reap, Cambodia, Dec. 2014.
- [13] J. Pang, G. Cheung, A. Ortega, and O. C. Au, "Optimal graph Laplacian regularization for natural image denoising," in *IEEE Int. Conf. Acoustics, Speech and Signal Processing*, Brisbane, Australia, Apr. 2015.
- [14] P. Wan, G. Cheung, D. Florencio, C. Zhang, and O. C. Au, "Image bit-depth enhancement via maximum-a-posteriori estimation of graph AC component," in *IEEE Int. Conf. Image Processing*, Paris, France, Oct. 2014.
- [15] F. K. Chung, *Spectral graph theory*. Providence, RI, USA: AMS, 1997.
- [16] D. A. Spielman, Lecture 2 of Spectral Graph Theory and its Applications, Dept. Comput. Sci., Yale Univ., New Haven, CT, USA, Sep. 2004.
- [17] U. Luxburg, "A tutorial on spectral clustering," *Statist. Comput.*, vol. 17, no. 4, pp. 395–416, Dec. 2007.
- [18] H. Chang, M. Ng, and T. Zeng, "Reducing artifacts in JPEG decompression via a learned dictionary," *IEEE Trans. Signal Process.*, vol. 62, no. 3, pp. 718–728, Feb. 2014.
- [19] K. Bredies and M. Holler, "Artifact-free decompression and zooming of JPEG compressed images with total generalized variation," *Comput. Vis. Imag. Comput. Graph.: Theory Applicat.*, vol. 359, pp. 242–258, 2013.