FAST & ROBUST IMAGE INTERPOLATION USING GRADIENT GRAPH LAPLACIAN REGULARIZER

Fei Chen*, Gene Cheung†, Xue Zhang†

*College of Mathematics and Computer Science, Fuzhou University, Fuzhou, China
†Dept of EECS, York University, Toronto, Canada

ABSTRACT

In the graph signal processing (GSP) literature, it has been shown that signal-dependent graph Laplacian regularizer (GLR) can efficiently promote piecewise constant (PWC) signal reconstruction for various image restoration tasks. However, for planar image patches, like total variation (TV), GLR may suffer from the well-known “staircase” effect. To remedy this problem, we generalize GLR to gradient graph Laplacian regularizer (GGLR) that provably promotes piecewise planar (PWP) signal reconstruction for the image interpolation problem—a 2D grid with random missing pixels that requires completion. Specifically, we first construct two higher-order gradient graphs to connect local horizontal and vertical gradients. Each local gradient is estimated using structure tensor, which is robust using known pixels in a small neighborhood, mitigating the problem of larger noise variance when computing gradient of gradients. Moreover, unlike total generalized variation (TGV), GGLR retains the quadratic form of GLR, leading to an unconstrained quadratic programming (QP) problem per iteration that can be solved quickly using conjugate gradient (CG). We derive the means-square-error minimizing weight parameter for GGLR, trading off bias and variance of the signal estimate. Experiments show that GGLR outperformed competing schemes in interpolation quality for severely damaged images at a reduced complexity.

Index Terms— Image interpolation, graph signal processing

1. INTRODUCTION

Due to limitations in image sensing, compression artifacts and transmission errors, acquired images are often imperfect with distorted or missing pixels. Image restoration is the task of recovering a pristine image from corrupted and/or partial observations. Appropriate signal priors are required to regularize an otherwise under-determined inverse problem. Classical methods like total variation (TV) [1] and sparse representation [2] take a model-based approach, where mathematical signal assumptions are made before data observations. In contrast, development in deep learning (DL) has led to purely data-driven regularizers that perform well on average [3, 4]. However, pure DL schemes require large training data and memory footprints, and perform poorly when statistics of training and testing data are mismatched [5, 6, 7]. In this paper, we propose a new model-based prior for image restoration that is both fast and robust. We leave the investigation of hybrid schemes that combine advantages of model-based and data-driven approaches [8, 9, 7] as future work.

Recent interest in graph signal processing (GSP) [10, 11]—analysis of discrete signals residing on irregular data kernels described by graphs—has resulted in graph spectral image restoration algorithms for denoising, deblurring, contrast enhancement, etc [12, 13, 14]. Extending previous model-based approaches, GSP-based schemes interpret an image as a signal on a suitably defined graph G, and assume x is smooth or bandlimited with respect to (w.r.t.) G. In particular, the signal-dependent graph Laplacian regularizer (SDGLR or GLR for short) [12, 15] quantifies smoothness of graph signal x w.r.t. G, specified by graph Laplacian matrix L:

\[ x^T L(x) x = \sum_{i,j} w_{i,j}(x_i, x_j) (x_i - x_j)^2 \] (1)

where \( w_{i,j} \) is the weight of an edge connecting nodes i and j, and \( x_i \) and \( x_j \) are the pixel location and intensity, respectively. Note that \( w_{i,j} \) is signal-dependent—it is a Gaussian function of intensity difference \( |x_i - x_j| \)—and hence L(x) is a function of sought signal x. This means that each term in the sum in (1) is minimized when \( |x_i - x_j| \) is either very small (thus (\( x_i - x_j \))^2 is small) or very large (thus \( w_{i,j} \) is small). As a result, minimizing GLR promotes piecewise constant (PWC) signal reconstruction as proven in [12, 15].

However, like TV that also promotes PWC signal reconstruction, GLR (1) suffers from the well-known “staircase” effect for image patches with linearly changing intensity. Extending GLR, in this paper we propose a higher-order graph smoothness prior called gradient graph Laplacian regularizer (GGLR) for the image interpolation problem—a 2D grid has randomly missing pixels that require completion. Specifically, for a target image, we first construct two gradient graphs to connect local horizontal and vertical gradients. Each local gradient is estimated using structure tensor [16], which is robust using known pixels in a small neighborhood, thus alleviating the pitfall of large noise variance when computing gradient of gradients. Unlike GLR, we prove that GGLR promotes a more general piecewise planar (PWP) signal reconstruction.

Computationally, GGLR retains the simple quadratic form in (1), leading to an unconstrained quadratic programming (QP) problem per iteration when edge weights are fixed, solvable using fast numerical linear algebra methods like conjugate gradient (CG) [17]. Leveraging [18], we derive the means-square-error (MSE) minimizing weight parameter for GGLR, trading off bias and variance of the signal estimate. Experiments show that GGLR is both fast and robust, outperforming competitors in interpolation quality when a large number of pixels are missing at a reduced complexity.

Related Work: Low-complexity classical interpolation methods such as bilinear interpolation [19] are commonly used in consumer software, but are sub-optimal in general due to the lack of signal adaptivity in local neighborhoods. Edge-guided methods such as...
partial differential equations (PDE) [20] provide smooth interpolation, but perform poorly when missing pixels are considerable.

TV [1] was a popular image prior due to its simplicity in definition and available optimization algorithms in minimizing convex but non-differentiable $\ell_1$-norm, which has no closed-form solutions. Its generalization, total generalized variation (TGV) [21, 22], better handles the aforementioned staircase effect, but retains the non-differentiable $\ell_1$-norm. Like TGV, GGLR also promotes PWP signal reconstruction, but can be optimized efficiently as an unconstrained $\ell_2$-norm QP problem per iteration—this is the focus of our paper.

2. GRAPH CONSTRUCTION

2.1. Horizontal Gradient Graph

Given an image $X \in \mathbb{R}^{M \times N}$, we compute its horizontal gradient $G^h \in \mathbb{R}^{M \times (N-1)}$ as the intensity differences between horizontally neighboring pixels, i.e.,

$$G^h_{k, i} = X_{k, i+1} - X_{k, i}$$

for $1 \leq k \leq M$ and $1 \leq i \leq N - 1$. We rewrite $G^h$ in vector form $g^h = \text{vec}(G^h) \in \mathbb{R}^{MN-1 \times 1}$, where $\text{vec}()$ denotes the vectorization operator stacking columns of a matrix into a vector. Thus we have a matrix representation $g^h = F^h x$, where $x = \text{vec}(X)$ and $F^h \in \mathbb{R}^{MN \times MN}$ is the horizontal gradient operator.

Next, we define a horizontal gradient graph $G^h$ consisting of $M(N-1)$ nodes corresponding to elements of $g^h$. Similar to (2), we define the weight $w^h_{i,j}$ for an edge connecting nodes $i$ and $j$ as

$$w^h_{i,j} = \begin{cases} \exp\left(-\frac{|g^h_{i} - g^h_{j}|^2}{\sigma^2}\right) & j \in N^h_i \\ 0 & \text{o.w.} \end{cases}$$

where $N^h_i$ is a defined local neighborhood centered at gradient $i$. $\sigma > 0$ is a chosen parameter so that edge weights $w^h_{i,j}$ are well distributed between 0 and 1. Define next an adjacency matrix $W^h \in \mathbb{R}^{MN \times MN}$ where $W^h_{i,j} = w^h_{i,j}$. A graph $G^h$ is completely characterized by the adjacency matrix $W^h$.

2.1.1. GGLR for Horizontal Gradients

We next define the diagonal degree matrix $D^h$ with diagonal elements $D^h_{i,i} = \sum_j W^h_{i,j}$ and then define a graph Laplacian matrix $L^h \in \mathbb{R}^{MN \times MN}$ as $L^h = D^h - W^h \in \mathbb{R}^{MN \times MN}$. Given that the edge weights are non-negative in (4), one can prove that Laplacian $L^h$ is positive semi-definite (PSD) via the Gershgorin circle theorem [11]. Since edge weights are defined w.r.t. gradients $g^h$ computed from signal $x$, Laplacian $L^h(x)$ is a function of $x$. We now write GLR for gradient $g^h$

$$(g^h)^T L^h g^h = \sum_{i=1}^{M-1} \sum_{j=1}^{N-1} w^h_{i,j} (g^h_i - g^h_j)^2$$

$$= x^T (F^h)^T L^h F^h x = x^T L^h x$$

where $L^h$ is also PSD since $L^h$ is PSD. Thus, GLR for gradient $g^h$ can also be computed in the pixel domain using $x$ in quadratic form via matrix $L^h$. We call (6) the horizontal GGLR for signal $x$.

As an example, consider a 3-pixel row $x = [x_1 \ x_2 \ x_3]$. Using $F^h = [1 \ -1 \ 0; \ 0 \ 1 \ -1]$ results in gradient $g^h = F^h x = [x_1 - x_3 \ x_2 - x_3]^T$. Assuming $w^h_{i,j} = 1$, $\forall (i,j)$, for simplicity, the corresponding matrices $L^h$ and $L^h$ are:

$$L^h = \begin{bmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{bmatrix}$$

Observe that $L^h$ is a Laplacian corresponding to a signed graph $G$, where $w_{1,2} = w_{2,3} = 1$ and $w_{1,3} = -1$. See Fig. 1(a) for an illustration. Unlike previous signed graphs in GSP, $G$ is not balanced in general [23] and has no self-loops [24], yet Laplacian $L^h$ is guaranteed to be PSD by definition (6), without any eigenvalue shift [25]. Thus, GGLR constitutes a new usage of signed graphs in GSP.

2.2. Vertical Gradient Graph

Similarly, we define the gradient in the vertical direction $G^v \in \mathbb{R}^{(M-1) \times N}$ as $G^v_{k, j} = X_{k+1, j} - X_{k, j}$ for $1 \leq k \leq M - 1$ and $1 \leq j \leq N - 1$. We have a similar matrix representation in the vertical direction $g^v = F^v x$, where $F^v \in \mathbb{R}^{(M-1) \times MN}$ is the vertical gradient operator. The vertical GGLR is thus $x^T L^v x$, where $L^v = (F^v)^T L^h F^v$. $L^v$ is also PSD since $L^v$ is PSD.

3. GRAPH LAPLACIAN REGULARIZER

As shown in Fig. 1(b)-(d), natural images are often approximately PWP. To promote PWP reconstruction, we define GGLR $\Phi(x)$ for an image $x$ by combining the horizontal and vertical GGLRs as

$$\Phi(x) = x^T L^h x + x^T L^v x = x^T (L^h + L^v) x$$

We first show GGLR promotes PWP signal reconstruction in 1D.

3.1. Piecewise Planar Signal Reconstruction

For simplicity, consider a row of $N$ pixels and a corresponding line graph with graph Laplacian $L^h = (F^h)^T L^h F^h$. Gradient operator $F^h$ in this 1D case is a $N - 1$-by-$N$ Toeplitz matrix:

$$F^h = \begin{bmatrix} 1 & -1 & 0 & \ldots \\ 0 & 1 & -1 & 0 & \ldots \\ \vdots & \ddots & \ddots & \ddots \end{bmatrix}$$

We see that the constant vector $1$ is in the null space of $F^h$, i.e., $F^h 1 = 0$. This means $v_1 = 1/\sqrt{N}$ is an eigenvector of $L^h$ corresponding to eigenvalue $0$, i.e., $L^h v_1 = (F^h)^T L^h F^h v_1 = 0$.

Consider next a linearly decreasing vector $x$, i.e.,

$$x_{i+1} = x_i - \delta, \quad 1 \leq i \leq N - 1$$

where $\delta > 0$ is a constant. Here, $F^h x = \delta 1$, and thus $L^h x = (F^h)^T L^h F^h x = \delta (F^h)^T L^h 1 = 0$, since a constant vector is the
first eigenvector for graph Laplacian $L^h$ corresponding to the smallest eigenvalue $0$ [10]. Define $v_2 = x/\|x\|_2$. Thus, any linear combination $y = c_1v_1 + c_2v_2$ for constants $c_i$ (i.e., linear signal of any slope) will also evaluate to $L^h y = 0$. Thus, GGLR $x^+ L^h x$, first and foremost, promotes linear signal reconstruction.

Consider next a $N$-pixel row consists of two linear pieces, of lengths $K$ and $N - K$ and slopes $\delta_1$ and $\delta_2$ respectively, separated by a large $\Delta$

\[
x_{i+1} = \begin{cases} 
  x_i - \delta_1 & \text{if } 1 \leq i \leq K - 1 \\
  x_i - \Delta & \text{if } i = K \\
  x_i - \delta_2 & \text{if } K + 1 \leq i \leq N \end{cases}
\]

(11)

where $|\Delta| \gg |\delta_1|, |\delta_2|$. Neighboring $g_h$’s within each linear piece will compute to the same value, and thus $|g_{i+1} - g_i| = 0$. Adjacent $g_h$’s across the piece boundary will compute to $|g_{i+1} - g_i| = |\Delta - \delta_1| \gg 0$. By edge weight definition (4), weight $w_{i+1,i} \approx 0$, and thus the corresponding term in (5) is also $\approx 0$. Thus, we conclude that signal $x$ that minimizes GGLR for 1D is linear or piecewise linear.

4. IMAGE INTERPOLATION USING GGLR

4.1. Gradient Estimation using Structure Tensor

Computation of edge weights in the gradient graph using (4) requires the difference of gradients $g_h - g_r = x_{i+1} - x_i - x_{i+1} + x_i$. If each pixel $x_i$ is corrupted by zero-mean additive noise with variance $\sigma^2$, then variable $g_h - g_r$ suffers from four times the noise variance $4\sigma^2$. Hence, a robust estimate of gradient $g_h$ is essential.

For the targeted image interpolation problem, we compute local gradients $g_h$ reliably using structure tensor [16]. Structure tensor $S(p)$ at a 2D location $p$ is a $2 \times 2$ matrix computed using observable horizontal and vertical gradients, $g_h^p$ and $g_r^p$ in a window, i.e.,

\[
S(p) = \begin{bmatrix}
\sum_{r \in U} \frac{1}{|U|} (g_h^r)^2 & \sum_{r \in U} \frac{1}{|U|} g_h^r g_r^p \\
\sum_{r \in U} \frac{1}{|U|} g_h^r g_r^p & \sum_{r \in U} \frac{1}{|U|} (g_r^p)^2
\end{bmatrix}
\]

(12)

where $U$ is a set of displacement vectors for the chosen window (e.g., $U = \{(0, 1), (0, -1), (1, 0), (-1, 0)\}$ for a window containing displacement vectors to the four closest neighbors on a 2D grid). We discount any gradient terms $g_h^p g_r^p$ in the sums in (12) that cannot be computed due to missing pixel(s).

Upon obtaining $S(p)$ in (12), we compute the eigenvector $\mathbf{e}$ corresponding to the larger of two eigenvalues. $\mathbf{e}$ is the dominant gradient of a patch center at $p$. We then project $\mathbf{e}$ into its $x$- and $y$-components as estimates of horizontal and vertical gradients, $g_h^p$ and $g_r^p$, at $p$. See Fig. 2 for an illustration.

4.2. Problem Formulation and Optimization

We now formulate the image interpolation problem using GGLR (8). Denote by $y \in \mathbb{R}^N$ an observation vector containing the $K$ known pixels with index set $K$ in an $M \times N$ image $X$. Denote by $H \in \{0, 1\}^{K \times MN}$ a selection matrix that picks out the $K$ known pixels from vectorized $x = \text{vec}(X) \in \mathbb{R}^{MN}$, i.e.,

\[
H_{ij} = \begin{cases} 
  1 & \text{if } j \text{ is the } i\text{-th available index in } K \\
  0 & \text{a.w.}
\end{cases}
\]

(13)

The optimal patch $x^*$ should respect observation $y$ while minimizes GGLR. Hence, the optimization objective is

\[
\min_{x} \|H x - y\|_2^2 + \mu x^\top (L^h + L^r) x
\]

(14)

where $\mu \geq 0$ is a parameter trading off the fidelity term and the GGLR prior. Note that $L^h$ and $L^r$ are signal-dependent functions of $x$ due to (4). As done in [12, 15] for GLR, we solve (14) via an iterative approach. For fixed $L^h$ and $L^r$, we first compute an optimal $x^*$. We then update edge weights using (4) in the two gradient graphs $g_h$ and $g_r$ having computed gradients $g_h^p$ and $g_r^p$ from solution $x^*$.

We repeat these two steps until convergence.

For fixed $L^h$ and $L^r$, (14) is an unconstrained QP problem with a closed-form solution:

\[
(H^\top H + \mu (L^h + L^r)) x^* = H^\top y.
\]

(15)

Coefficient matrix $B = H^\top H + \mu (L^h + L^r)$ is positive definite (PD) for sufficiently large $K$. For simplicity, we prove only the 1D case when $x$ is a row of pixels, and $B$ is PD when $K > 1$. As shown in Section 3.1, $y^\top L^h y = 0$ implies that $y$ is a linear signal. $H^\top H$ is a diagonal matrix with 1’s and 0’s along its diagonal. Hence, for $K > 1$, quadratic term for matrix $H^\top H$ is $y^\top H^\top H y = \sum_{i \in K} y_i^2 > 0$, since linear signal $y \neq 0$ can have at most one entry $y_i = 0$. Thus, quadratic terms for $H^\top H$ and $L^h$ cannot both evaluate to 0 using $y$. Given both $H^\top H$ and $L^h$ are PSD, we conclude that $x^* (H^\top H + L^h) x^* > 0 \forall x \neq 0$, and therefore $H^\top H + L^h$ is PD. Similarly, $H^\top H + L^h$ is PD for $\mu > 0$.

Given $B$ is symmetric, sparse and PD, (15) can be solved efficiently using conjugate gradient (CG) [17], without computing matrix inverse, in $O(cMN)$, where $c$ is the iteration number in CG.

4.3. Choosing Tradeoff Parameter $\mu$

Parameter $\mu$ in (14) must be carefully chosen for best performance. Analysis of $\mu$ for GLR-based signal denoising [18] showed that the best $\mu$ minimizes the mean square error (MSE) by optimally trading off bias square and variance of the estimate with its variance. Here, we derive a new variant from Corollary 1 in [18] to compute $\mu$ for (14).

For simplicity, consider again a row of pixels where $K = NM$. Thus $H = I$, and $B = I + \mu L^h$. Assume that observation $y = x^0 + z$ is corrupted by zero-mean iid noise $z$ with covariance matrix $\sigma^2 I$. Denote by $(\lambda_1, \nu_1)$ the $i$-th eigen-pair of matrix $L^h$. Since $\lambda_1 = \lambda_2 = 0$, Theorem 2 for MSE in [18] can be restated as

\[
\text{MSE}(\mu) = \sum_{i=3}^{K} q_i^2 (\nu_i^2 \sigma^2) + \sigma^2 \sum_{i=1}^{K} h_i^2
\]

(16)

where $x^0 = x^r - \frac{1}{\text{vec}(L^h)\lambda_1}, q_i = \frac{1}{1 + \lambda_i},$ and $h_i = \frac{1}{1 + \lambda_i}$. The two terms in (16) correspond to bias square and variance of estimate $x^*$. Suppose now that ground truth $x^0 = x^r + x_p$ is a linear signal $x^r_p$ defined in (10) plus zero-mean iid perturbation $x_p$ with covariance matrix $\sigma^2 I$. If gradient graph $g^h$ is computed from $x^r_p$, then $\nu_1 = \nu^r_1$. We compute expectation $\mathbb{E}(\nu_i^2 x^0)^2$ for $i \geq 3$:

\[
\mathbb{E}(\nu_i^2 x^0)^2 = tr \left( \nu_i \nu_i^\top (x^r_p(x^r_p)^\top + \sigma^2 I) \right) = \sigma^2 \nu^2.
\]

(17)
Fig. 3. Visual and PSNR comparison for patterned missing pixels. (a) Corrupted images. (b-c) Interpolation by SDGLR and GGLR.

Table 1. Interpolation results by different methods on 50 images with ≥ 90% randomly missing pixels. GGLR$^4$ (GGLR$^2$) uses 4-connected (2-connected) horizontal and vertical gradient graphs.

| Methods | SDGLR | CSC | IRCNN | IDBP | TGV | GSC | GGLR$^4$ | GGLR$^2$ |
|---------|-------|-----|-------|------|-----|-----|----------|----------|
| PSNR(dB) | 21.55 | 20.92 | 22.03 | 22.60 | 22.16 | 22.57 | 22.23   | 21.90   |
| SSIM    | 0.628 | 0.606 | 0.651 | 0.680 | 0.667 | 0.685 | 0.683   | 0.670   |

95% missing pixels

| Methods | SDGLR | CSC | IRCNN | IDBP | TGV | GSC | GGLR$^4$ | GGLR$^2$ |
|---------|-------|-----|-------|------|-----|-----|----------|----------|
| PSNR(dB) | 19.87 | 18.41 | 16.75 | 20.61 | 20.42 | 20.38 | 20.46   | 20.24   |
| SSIM    | 0.529 | 0.486 | 0.477 | 0.584 | 0.571 | 0.570 | 0.587   | 0.575   |

98% missing pixels

| Methods | SDGLR | CSC | IRCNN | IDBP | TGV | GSC | GGLR$^4$ | GGLR$^2$ |
|---------|-------|-----|-------|------|-----|-----|----------|----------|
| PSNR(dB) | 18.02 | 12.21 | 9.95  | 18.11 | 18.32 | 18.09 | 18.37   | 18.17   |
| SSIM    | 0.435 | 0.263 | 0.229 | 0.479 | 0.477 | 0.456 | 0.485   | 0.471   |

99% missing pixels

| Methods | SDGLR | CSC | IRCNN | IDBP | TGV | GSC | GGLR$^4$ | GGLR$^2$ |
|---------|-------|-----|-------|------|-----|-----|----------|----------|
| PSNR(dB) | 16.89 | 9.01  | 7.60  | 16.54 | 17.07 | 16.85 | 17.11   | 16.90   |
| SSIM    | 0.386 | 0.137 | 0.118 | 0.416 | 0.420 | 0.393 | 0.423   | 0.411   |

We now restate Corollary 1 in [18] as an upper bound for MSE:

$$\text{MSE}(\mu) \leq \frac{(K-2)\sigma_p^2}{(1 + \frac{1}{\mu \lambda_3}^2) + \frac{1}{\mu \lambda_3} + 2} \sigma_o^2. \quad (18)$$

The MSE bound (18) can be minimized by taking the derivative w.r.t. $\mu$ and setting it to 0.

5. EXPERIMENTS

We first describe parameter setting in our experiments. $N_i$ was set to 2-neighborhood (4-neighborhood), for 2-connected (4-connected) gradient graphs. $\sigma$ in (4) was set to 0.68. Window $U$ in (12) was set to $5 \times 5$. Based on MSE analysis in Section 4.3, $\mu$ was set to 0.01.

We first visually compare the performance of GGLR with SDGLR. As shown in Fig. 3, interpolation of SDGLR appeared over-smoothed and inconsistent with surrounding textures. In contrast, GGLR better preserved image gradients, resulting in more natural reconstruction that was consistent with surrounding textures.

We compare against state-of-the-art image interpolation methods: SDGLR, TGV [22], CSC [26], IRCNN [27], IDBP [28] and GSC [29], where IRCNN is a DL based method. The first 50 images from Weizmann Horse Database [30] were used for our experiments. The average PSNR and SSIM of the reconstructed images are listed in Table 1. We observe that GGLR outperformed competing methods in general when the fraction of missing pixels was large. A typical visual comparison is shown in Fig. 4. GGLR preserved image contours and mitigated blocking effects, observed in interpolation by sparse representation (e) and (g). GGLR achieved the highest PSNR and SSIM when the fraction of missing pixels is above 98%.

Table 2 reports runtime of different methods for a $128 \times 128$ image. All experiments were run under the Matlab2015b environment on a laptop with Intel Core i5-8365U CPU of 1.60GHz. TGV employed a primal-dual splitting method [31] for $\ell_2-\ell_1$ norm minimization, which required a large number of iterations until convergence, especially when the fraction of missing pixels was large. In contrast, GGLR employing line graphs required roughly 1sec.

6. CONCLUSION

We propose a high-order signal-dependent gradient graph Laplacian regularizer (GGLR) that promotes piecewise planar (PWP) signal reconstruction in 2D images. Unlike total generalized variation (TGV), GGLR is in quadratic form, so that in each iteration, when edge weights are fixed, an unconstrained quadratic programming (QP) problem can be solved efficiently using conjugate gradient (CG). Image interpolation experiments show that GGLR outperformed competitors when fraction of missing pixels was large.
7. REFERENCES

[1] A. Chambolle and P.-L. Lions, “Image recovery via total variation minimization and related problems,” Numerische Mathematik, vol. 76, no. 2, pp. 167–188, 1997.

[2] D. L. Donoho and M. Elad, “Optimally sparse representation in general (nonorthogonal) dictionaries via ́1 minimization,” Proceedings of the National Academy of Sciences, vol. 100, no. 5, pp. 2197–2202, 2003.

[3] L. Zhang and W. Zuo, “Image restoration: From sparse and low-rank priors to deep priors [lecture notes],” IEEE Signal Processing Magazine, vol. 34, no. 5, pp. 172–179, 2017.

[4] S.-A. Bigdeli, M. Zwicker, P. Favaro, and M. Jin, “Deep mean-shift priors for image restoration,” in Advances in Neural Information Processing Systems, 2017, pp. 763–772.

[5] X. Peng, J. Hoffman, X. Y. Stella, and K. Saenko, “Fine-to-coarse knowledge transfer for low-res image classification,” in 2016 IEEE International Conference on Image Processing (ICIP). IEEE, 2016, pp. 3683–3687.

[6] H. Touvron, A. Vedaldi, M. Douze, and H. Jégou, “Fixing the train-test resolution discrepancy,” in Advances in neural information processing systems, 2019, pp. 8252–8262.

[7] H. Vu, G. Cheung, and Y. C. Eldar, “Unrolling of deep graph total variation for image denoising,” in IEEE International Conference on Acoustics, Speech and Signal Processing, Toronto, Canada, June 2017.

[8] J. Zeng, J. Pang, W. Sun, and G. Cheung, “Deep graph Laplacian regularization for robust denoising of real images,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2019, pp. 0–0.

[9] W.-T. Su, G. Cheung, R. Wildes, and C.-W. Lin, “Graph neural net using analytical graph filters and topology optimization for image denoising,” in IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2020, pp. 8464–8468.

[10] A. Ortega et al., “Graph signal processing: Overview, challenges, and applications,” Proceedings of the IEEE, vol. 106, no. 5, pp. 808–828, 2018.

[11] G. Cheung et al., “Graph spectral image processing,” Proceedings of the IEEE, vol. 106, no. 5, pp. 907–930, 2018.

[12] J. Pang and G. Cheung, “Graph Laplacian regularization for image denoising: Analysis in the continuous domain,” IEEE Transactions on Image Processing, vol. 26, no. 4, pp. 1770–1785, 2017.

[13] Y. Bai, G. Cheung, X. Liu, and W. Gao, “Graph-based blind image deblurring from a single photograph,” IEEE Transactions on Image Processing, vol. 28, no.3, pp. 1404–1418, 2019.

[14] X. Liu, G. Cheung, X. Ji, D. Zhao, and W. Gao, “Graph-based joint dequantization and contrast enhancement of poorly lit JPEG images,” IEEE Transactions on Image Processing, vol. 28, no.3, pp. 1205–1219, March 2019.

[15] X. Liu, G. Cheung, X. Wu, and D. Zhao, “Random walk graph Laplacian based smoothness prior for soft decoding of JPEG images,” IEEE Transactions on Image Processing, vol. 26, no.2, pp. 509–524, February 2017.

[16] H. Knutsson, C.-F. Westin, and M. Andersson, “Representing local structure using tensors II,” in Scandinavian Conference on Image Analysis. Springer, 2011, pp. 545–556.

[17] O. Axelsson and G. Lindskog, “On the rate of convergence of the preconditioned conjugate gradient method,” Numerische Mathematik, vol. 48, no. 5, pp. 499–523, 1986.

[18] P.-Y. Chen and S. Liu, “Bias-variance tradeoff of graph Laplacian regularizer,” IEEE Signal Processing Letters, vol. 24, no. 8, pp. 1118–1122, 2017.

[19] WH Press, SA Teukolsky, WT Vetterling, and BP Flannery, “Numerical recipes in C: The art of scientific computing 2nd ed. 1992 cambridge univ,” PressCambridge, UK.

[20] G. B Folland, Introduction to partial differential equations, vol. 102, Princeton university press, 1995.

[21] K. Bredies, K. Kunisch, and T. Pock, “Total generalized variation,” SIAM Journal on Imaging Sciences, vol. 3, no. 3, pp. 492–526, 2010.

[22] K. Bredies and M. Holler, “A TGV-based framework for variational image decomposition, zooming, and reconstruction. Part I: Analytics,” SIAM Journal on Imaging Sciences, vol. 8, no. 4, pp. 2814–2850, 2015.

[23] C. Yang, G. Cheung, and W. Hu, “Signed graph metric learning via Gershgorin disc perfect alignment,” (accepted to) IEEE Transactions on Pattern Analysis and Machine Intelligence, June 2021.

[24] W.-T. Su, G. Cheung, and C.-W. Lin, “Graph Fourier transform with negative edges for depth image coding,” in IEEE International Conference on Image Processing, Beijing, China, September 2017.

[25] G. Cheung, W.-T. Su, Y. Mao, and C.-W. Lin, “Robust semi-supervised graph classifier learning with negative edge weights,” in IEEE Transactions on Signal and Information Processing over Networks, December 2018, vol. 4, no.4, pp. 712–726.

[26] E. Zisselman, J. Sulam, and M. Elad, “A local block coordinate descent algorithm for the CSC model,” in The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2019.

[27] K. Zhang, W. Zuo, S. Gu, and L. Zhang, “Learning deep CNN denoiser prior for image restoration,” in IEEE Conference on Computer Vision and Pattern Recognition, 2017.

[28] T. Tirer and R. Giryes, “Image restoration by iterative denoising and backward projections,” IEEE Transactions on Image Processing, vol. 28, no. 3, pp. 1220–1234, 2019.

[29] Z. Zha, X. Yuan, B. Wen, J. Zhou, J. Zhang, and C. Zhu, “A benchmark for sparse coding: When group sparsity meets rank minimization,” IEEE Transactions on Image Processing, vol. 29, pp. 5094–5109, 2020.

[30] E. Borenstein and S. Ullman, “Class-specific, top-down segmentation,” in European Conference on Computer Vision (ECCV), 2002.

[31] L. Condat, “A primal-dual splitting method for convex optimization involving Lipschitzian, proximable and linear composite terms,” in J. Optimization Theory and Applications, 2013, vol. 158, pp. 460–479.