OPTIMAL GRAPH LAPLACIAN REGULARIZATION FOR NATURAL IMAGE DENOISING

Jiahao Pang [*], *Gene Cheung* [†], *Antonio Ortega* [$], *Oscar C. Au* [*]

[*] The Hong Kong University of Science and Technology, Hong Kong, China
[†] National Institute of Informatics, Tokyo, Japan, [$] University of Southern California, USA

ABSTRACT

Image denoising is an under-determined problem, and hence it is important to define appropriate image priors for regularization. One recent popular prior is the graph Laplacian regularizer, where a given pixel patch is assumed to be smooth in the graph-signal domain. The strength and direction of the resulting graph-based filter are computed from the graph's edge weights. In this paper, we derive the optimal edge weights for local graph-based filtering using gradient estimates from non-local pixel patches that are self-similar. To analyze the effects of the gradient estimates on the graph Laplacian regularizer, we first show theoretically that, given graph-signal $\mathbf{h}^D$ is a set of discrete samples on continuous function $h(x, y)$ in a closed region Ω, graph Laplacian regularizer $(\mathbf{h}^D)^{\mathrm{T}}\mathbf{L}\mathbf{h}^D$ converges to a continuous functional S_Ω integrating gradient norm of h in metric space $\mathbf{G}$ — *i.e.*, $(\nabla h)^{\mathrm{T}}\mathbf{G}^{-1}(\nabla h)$ — over Ω. We then derive the optimal metric space $\mathbf{G}^\star$: one that leads to a graph Laplacian regularizer that is discriminant when the gradient estimates are accurate, and robust when the gradient estimates are noisy. Finally, having derived $\mathbf{G}^\star$ we compute the corresponding edge weights to define the Laplacian $\mathbf{L}$ used for filtering. Experimental results show that our image denoising algorithm using the per-patch optimal metric space $\mathbf{G}^\star$ outperforms non-local means (NLM) by up to 1.5 dB in PSNR.

Index Terms— graph Laplacian regularization, metric space, image denoising, inverse imaging problem

1. INTRODUCTION

Image denoising [1] seeks the original signal $\mathbf{x}$ given observed signal $\mathbf{y} = \mathbf{x} + \mathbf{e}$ corrupted by additive noise $\mathbf{e}$. It is an under-determined problem, and thus appropriate definition of image priors to regularize the problem is important. Among proposed image priors in the literature such as total variation (TV) [2] and auto-regressive prior [3], one recent popular prior is the *graph Laplacian regularizer* [4, 5, 6]. Leveraging on recent advance in graph signal processing (GSP) [7], graph Laplacian regularizer states that a desirable image patch $\mathbf{x}$ in vector form induces a small value $S_{\mathcal{G}}(\mathbf{x}) = \mathbf{x}^{\mathrm{T}}\mathbf{L}\mathbf{x}$, where $\mathbf{L} = \mathbf{D} - \mathbf{A}$ is the *graph Laplacian matrix* for a well-defined combinatorial graph $\mathcal{G}$ with degree and adjacency matrices $\mathbf{D}$ and $\mathbf{A}$, and whose vertices correspond to pixels in patch $\mathbf{x}$. Using such regularization term, the denoising problem can now be formulated as:

$$\mathbf{x}^\star = \arg\min_{\mathbf{x}} \ \|\mathbf{y} - \mathbf{x}\|_2^2 \ + \ \lambda\, \mathbf{x}^{\mathrm{T}}\mathbf{L}\mathbf{x}. \tag{1}$$

Because (1) is an unconstrained quadratic optimization problem, the optimal solution $\mathbf{x}^\star$ can be derived in closed form [7]:

$$\mathbf{x}^\star = (\mathbf{I} + \lambda\mathbf{L})^{-1}\,\mathbf{y}, \tag{2}$$

where $\mathbf{I}$ denotes the identity matrix. One can thus interpret the solution $\mathbf{x}^*$ as a locally filtered version of noisy input $\mathbf{y}$ using linear filter $(\mathbf{I} + \lambda\mathbf{L})^{-1}$.

It is clear from (2) that the strength and direction of the resulting local filter depends on the edge weights that define adjacency matrix $\mathbf{A}$ (and subsequently Laplacian $\mathbf{L}$). In this paper, assuming that pixel patches with *similar gradients* recur throughout a natural image (akin to the image *self-similarity* assumption in *non-local means* (NLM) [8]), we derive optimal edge weights for local graph-based filtering using gradients estimated from a set of K non-local similar pixel patches. This means that the effectiveness of the graph Laplacian regularizer depends on the overall quality of the K gradient estimates. Compared to bilateral filtering [9], our filtering in (2) is also a local filter but the weights are computed using K non-local similar patches, resulting in more robust weight estimates. Yet unlike NLM [8], filtering of a pixel is still performed locally using its neighboring pixels, meaning local luminance characteristics (pixels in a local patch likely have similar luminance values assuming an intrinsic image model [10]) can be preserved.

To analyze the effects of the K non-local gradient estimates on the graph Laplacian regularizer, we first show theoretically that, given graph-signal $\mathbf{h}^D$ is a set of discrete samples on continuous function $h(x, y)$ in a closed region Ω, graph Laplacian regularizer $(\mathbf{h}^D)^{\mathrm{T}}\mathbf{L}\mathbf{h}^D$ converges to a continuous functional S_Ω integrating the gradient norm of h measured in *metric space* $\mathbf{G}$ — *i.e.*, $(\nabla h)^{\mathrm{T}}\mathbf{G}^{-1}(\nabla h)$ — over Ω. We then derive the optimal metric space $\mathbf{G}^\star$: one that leads to a graph Laplacian regularizer that is *discriminant* when the K gradient estimates are accurate, and *robust* when the K gradient estimates are noisy. Finally, having derived $\mathbf{G}^\star$ we compute the corresponding edge weights to define the Laplacian $\mathbf{L}$ used for filtering in (2). Experimental results show that our denoising algorithm using the per-patch optimal metric space $\mathbf{G}^\star$ outperforms NLM by up to 1.5 dB in PSNR.

The outline of the paper is as follows. We first show convergence of graph Laplacian regularizer to its corresponding continuous functional in Section 2. We then derive the optimal metric space and the corresponding graph Laplacian in Section 3. Experimentation and conclusion are presented in Section 4 and 5, respectively.

2. CONTINUOUS DOMAIN INTERPRETATION OF GRAPH LAPLACIAN REGULARIZATION

We first formally define a graph-signal and demonstrate the convergence of the graph Laplacian regularizer $S_{\mathcal{G}} = (\mathbf{h}^D)^{\mathrm{T}}\mathbf{L}\mathbf{h}^D$ to a continuous functional S_Ω. To gain better understanding of $S_{\mathcal{G}}$, we then present a detailed analysis of the functional S_Ω.

2.1. Graph Construction from Feature functions

Denote a bounded region on the 2D plane by $\Omega \subset \mathbb{R}^2$; we call Ω the *domain* in the sequel. In practice, Ω takes the shape of an image (or image patch) which is typically a rectangle (*e.g.*, Fig. 1). Let $\Gamma = \{\mathbf{s}_i = (x_i, y_i) \mid \mathbf{s}_i \in \Omega, 1 \le i \le M\}$ be a set of M *random samples* uniformly distributed on Ω. Since images are typically sam-

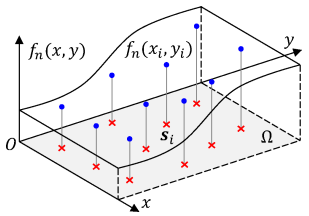


Fig. 1. Sampling the function f_n uniformly in domain Ω. Red crosses are sampling positions and blue dots are the samples.

pled uniformly on Ω in a 2D grid, samples in the set Γ are construed as pixel coordinates in this work.

Let $f_n(x,y) : \Omega \to \mathbb{R}$, $1 \le n \le N$, be continuous *feature functions* defined on domain Ω and can be freely chosen by users. For image denoising, f_n's can be chosen to be the intensity of the observed grayscale patch ($N = 1$), or R, G, and B channels of a noisy color image ($N = 3$), or K patches similar to the observed patch ($N = K$) due to self-similarity. Our work chooses f_n's optimally, which will be derived in Section 3. Sampling the feature functions $\{f_n\}_{n=1}^N$ at positions in Γ gives N vectors of length M,

$$\mathbf{f}_n^D = [f_n(x_1,y_1)\ f_n(x_2,y_2)\ \dots\ f_n(x_M,y_M)]^{\mathrm{T}}, \tag{3}$$

where $1 \le n \le N$, and superscript D means "discrete". Fig. 1 illustrates the sampling process of an example function f_n — a simple ramp in domain Ω. The red crosses are sampling positions in Γ. The blue dots are samples of f_n and collectively form vector $\mathbf{f}_n^D$.

For each sample location $\mathbf{s}_i \in \Gamma$, we construct a length N vector $\mathbf{v}_i$ ($1 \le i \le M$) with previously defined $\mathbf{f}_n^D$,

$$\mathbf{v}_i = [\mathbf{f}_1^D(i)\ \mathbf{f}_2^D(i)\ \dots\ \mathbf{f}_N^D(i)]^{\mathrm{T}}, \tag{4}$$

where $\mathbf{f}_n^D(i)$ is the i-th entry of $\mathbf{f}_n^D$. With vectors $\{\mathbf{v}_i\}_{i=1}^M$, we build a weighted neighborhood graph $\mathcal{G}$, where each sample (or pixel) $\mathbf{s}_i \in \Gamma$ is represented by a vertex V_i. The weight w_{ij} between two different vertices V_i and V_j is computed as

$$w_{ij} = (\rho_i \rho_j)^{-\gamma} \psi(d_{ij}). \tag{5}$$

The weighting kernel $\psi(\cdot)$ is a truncated Gaussian

$$\psi(d_{ij}) = \begin{cases} \exp\left(-\dfrac{d_{ij}^2}{2\epsilon^2}\right) & \text{if } |d_{ij}| \le r, \\ 0 & \text{otherwise,} \end{cases} \tag{6}$$

where the distance $d_{ij}^2 = \|\mathbf{v}_i - \mathbf{v}_j\|_2^2$, and the degree of V_i before normalization is $\rho_i = \sum_{j=1}^M \psi(d_{ij})$.

Under these settings, $\mathcal{G}$ is an r-neighborhood graph; *i.e.*, there is no edge connecting two vertices with distance greater than r. Here $r = \epsilon C_r$, and C_r is a constant. The parameter ϵ controls the sensitivity of the graph weights to the distances, and γ controls the normalization of the weights. Let the adjacency matrix of $\mathcal{G}$ be $\mathbf{A}$, where entry (i,j) of $\mathbf{A}$ is w_{ij}. The degree matrix of $\mathcal{G}$ is a diagonal matrix $\mathbf{D}$ with entry (i,i) computed as $\sum_{j=1}^M w_{ij}$. The unnormalized graph Laplacian $\mathbf{L}$ is simply $\mathbf{L} = \mathbf{D} - \mathbf{A}$. Note that graphs employed in many recent works (*e.g.*, [4, 5, 11, 12]) are special cases of our more generally defined graph $\mathcal{G}$.

2.2. Convergence of The Graph Laplacian Regularizer

Suppose $h(x,y) : \Omega \to \mathbb{R}$ is yet another continuous function defined on domain Ω. Sampling h at positions in Γ leads to its discretized version, $\mathbf{h}^D = [h(x_1,y_1)\ h(x_2,y_2)\ \dots\ h(x_M,y_M)]^{\mathrm{T}}$. The graph Laplacian $\mathbf{L}$ obtained in Section 2.1 induces the *graph Laplacian regularizer* $S_{\mathcal{G}}(\mathbf{h}^D) = (\mathbf{h}^D)^{\mathrm{T}} \mathbf{L} \mathbf{h}^D$.

The continuous counterpart of regularizer $S_{\mathcal{G}}(\mathbf{h}^D)$ is given by a functional $S_\Omega(h)$ on domain Ω,

$$S_\Omega(h) = \iint_\Omega (\nabla h)^{\mathrm{T}} \mathbf{G}^{-1} (\nabla h) \left(\sqrt{\det \mathbf{G}}\right)^{2\gamma-1} dxdy, \tag{7}$$

where $\nabla h = [\partial_x h\ \ \partial_y h]^{\mathrm{T}}$ is the gradient of continuous function h. $\mathbf{G}$ is a 2×2 matrix given by

$$\mathbf{G} = \sum_{n=1}^N \nabla f_n \cdot (\nabla f_n)^{\mathrm{T}}. \tag{8}$$

$\mathbf{G}$, a matrix-valued function of location (x,y), is also the *structure tensor* [13] of the gradients $\{\nabla f_n\}_{n=1}^N$. We see that feature functions $\{f_n\}_{n=1}^N$ determine the functional S_Ω and the graph Laplacian regularizer $S_{\mathcal{G}}$. We can now declare the following theorem:

Theorem 1 (Convergence of $S_{\mathcal{G}}$)**.** *Under mild conditions for ϵ, functions $\{f_n\}_{n=1}^N$ and h as stated in* [14]*,*

$$\lim_{\substack{M\to\infty \\ \epsilon\to0}} \frac{M^{2\gamma-1}}{\epsilon^{4(1-\gamma)}(M-1)} S_{\mathcal{G}}(\mathbf{h}^D) \sim S_\Omega(h), \tag{9}$$

where "$\sim$" means there exists a constant depending on Ω, C_r, and γ, such that equality holds.

In other words, as the number of samples M increases and the neighbourhood size $r = \epsilon C_r$ shrinks, graph Laplacian regularizer $S_{\mathcal{G}}$ approaches the continuous functional S_Ω. We proved Theorem 1 in [12], by viewing a graph as discrete approximation of a *Riemannian manifold*. [1]

2.3. Metric Space in Continuous Domain

Convergence of the graph Laplacian regularizer $S_{\mathcal{G}}$ to the continuous functional S_Ω allows us to understand the behaviour of $S_{\mathcal{G}}$ via analysis of S_Ω. In (7), the quadratic term $(\nabla h)^{\mathrm{T}} \mathbf{G}^{-1} (\nabla h)$ measures the gradient norm of h in a metric space defined by matrix $\mathbf{G}$. From (8), $\mathbf{G}$'s eigenvectors and eigenvalues capture the statistics of $\{\nabla f_n\}_{n=1}^N$. Fig. 2 shows three examples of different metric spaces in gradient coordinates for some location $(x,y) \in \Omega$. The green dots are gradients $\{\nabla f_n\}_{n=1}^N$. The eigenvector corresponding to the largest eigenvalue of $\mathbf{G}$ has direction l, which passes through the origin and *aligns* with the center of $\{\nabla f_n\}_{n=1}^N$. The ellipses are isolevel lines (norm balls) of the metric spaces, and the flatness of their shapes reflects how concentrated $\{\nabla f_n\}_{n=1}^N$ are.

Fig. 2 also shows different scenarios when S_Ω is used as a regularization term to recover the ground truth continuous patch (denoted as h_t). We mark the ground truth gradient — defined as $\mathbf{g} = \nabla h_t$ — with the red crosses in Fig. 2. We see that, though both the metric spaces in Fig. 2(a) and Fig. 2(b) skew toward $\mathbf{g}$, the one in Fig. 2(b) is more skewed or *discriminant*; *i.e.*, a small Euclidean distance away from the ground truth gradient $\mathbf{g}$ in a direction orthogonal to l will result in a large metric distance. This is *desirable* for a regularization term to distinguish between good pixel patch candidates (close to ground truth) and bad candidates (far from ground truth).

However, if the gradients $\{\nabla f_n\}_{n=1}^N$ concentrated in a location far from ground truth $\mathbf{g}$—resulting in a skewed metric space that is not aligned with $\mathbf{g}$—then the metric space is discriminant but incorrect (as shown in Fig. 2(c)); *i.e.,* bad pixel patch candidates would have smaller regularization terms than good candidates, which is *undesirable*. Therefore, *one should design a discriminant metric space*

[1] In [12], Theorem 1 is proved by applying the result of [14]. Different from [12], in this work the spatial coordinates are not included in $\mathbf{v}_i$, leading to a more general graph $\mathcal{G}$, but the proof in [12] remains valid.

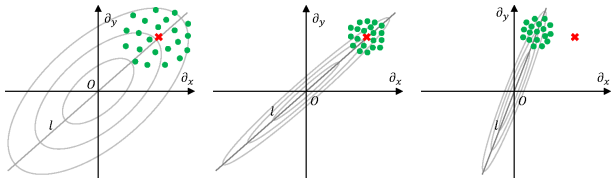


Fig. 2. At a position $(x, y) \in \Omega$, different arrangements of gradients $\{\nabla f_n\}_{n=1}^N$ establish different metric spaces for regularization.

only to the extent that estimates of $\mathbf{g}$ *are reliable*. We discuss the procedure to obtain the optimal metric space in the next section.

We first establish an ideal metric space when ground truth $\mathbf{g}$ is known. Specifically, $\forall (x, y) \in \Omega$, we seek for a matrix $\mathbf{G}$ to define a metric space, so that S_Ω is *discriminant* with respect to $\mathbf{g}$,

$$\mathbf{G}_0(\mathbf{g}) = \mathbf{g}\mathbf{g}^{\mathrm{T}} + \alpha \mathbf{I}, \tag{10}$$

where $\alpha > 0$ is a small constant. Given definition in (10), $\mathbf{G}_0(\mathbf{g})$ has two eigenvalues $\|\mathbf{g}\|_2^2 + \alpha$ and α. Moreover, (10) can be rewritten as

$$\mathbf{G}_0(\mathbf{g}) = \left(1 + \frac{\alpha}{\|\mathbf{g}\|_2^2}\right) \cdot \mathbf{g}\mathbf{g}^{\mathrm{T}} + \frac{\alpha}{\|\mathbf{g}\|_2^2} \cdot \mathbf{g}_\perp \mathbf{g}_\perp^{\mathrm{T}}, \tag{11}$$

where $\mathbf{g}_\perp$ is a vector orthogonal to $\mathbf{g}$, *e.g.*, $\mathbf{g}$ rotated by $\pi/2$ clockwise. The skewness of the metric space can be adjusted using α (small α means a more skewed metric space).

However, in image denoising the ground-truth gradient $\mathbf{g}$ is usually unknown. We discuss how to find an optimal metric space for image denoising in the next section.

3. IMAGE DENOISING WITH OPTIMAL GRAPH LAPLACIAN REGULARIZATION

We first introduce our noise model in gradient domain. Given a set of noisy gradient observations, we then derive the optimal metric space in the *minimum mean square error* (MMSE) sense. From the optimal metric space we derive the corresponding feature functions, leading to the optimal edge weights for the graph modeling pixels in the target patch. Our work adopts a patch-based recovery framework as done in [8, 11, 12]. We further assume the input image is corrupted by independent and identically distributed (i.i.d.) additive white Gaussian noise (AWGN).

3.1. Noise Modeling of the Gradients

Given a $\sqrt{M} \times \sqrt{M}$ noisy image patch $\mathbf{p}_0 \in \mathbb{R}^M$, one can identify a set of $K-1$ patches on the noisy image that are similar in terms of gradients. Together with $\mathbf{p}_0$, the K patches $\{\mathbf{p}_k\}_{k=0}^{K-1}$ are collectively called a *cluster* in the sequel. On patch $\mathbf{p}_k$, the 2D gradient at pixel i is denoted as $\mathbf{g}_k^i$. With all the noisy gradient observations at some pixel i, we seek the optimal metric space for this pixel. For simplicity, hereafter the superscript i is neglected. We model the noisy gradients $\{\mathbf{g}_k\}_{k=0}^{K-1}$ as

$$\mathbf{g}_k = \mathbf{g} + \mathbf{e}_k, \ 0 \le k \le K-1, \tag{12}$$

where $\mathbf{g}$ is the unknown ground-truth gradient on $\mathbf{p}_0$ at pixel i. $\{\mathbf{e}_k\}_{k=0}^{K-1}$ are independent noise terms, which follow 2D Gaussian distribution with zero-mean and covariance matrix $\sigma_e^2 \mathbf{I}$. $\{\mathbf{e}_k\}_{k=0}^{K-1}$ are the outcome of the i.i.d. AWGN process[2], and the formation noise introduced in the image formation process. Hence the probability density function (PDF) of $\mathbf{g}_k$ given ground truth $\mathbf{g}$ is

$$Pr(\mathbf{g}_k | \mathbf{g}) = \frac{1}{2\pi\sigma_e^2} \exp\left(-\frac{1}{2\sigma_e^2} \|\mathbf{g} - \mathbf{g}_k\|_2^2\right). \tag{13}$$

Our work assumes that the noise statistics change slowly, so that the variance σ_e^2 is a constant for a cluster $\{\mathbf{p}_k\}_{k=0}^{K-1}$.

3.2. Seeking for the Optimal Metric Space

Given the noisy gradients $\{\mathbf{g}_k\}_{k=0}^{K-1}$, we seek the optimal metric space for pixel i in the MMSE sense. Specifically, we formulate the following minimization problem to find the optimal matrix $\mathbf{G}^*$ given $\{\mathbf{g}_k\}_{k=0}^{K-1}$:

$$\mathbf{G}^\star = \arg\min_{\mathbf{G}} \iint_\Delta \|\mathbf{G} - \mathbf{G}_0(\mathbf{g})\|_F^2 \cdot Pr\left(\mathbf{g} \,\middle|\, \{\mathbf{g}_k\}_{k=0}^{K-1}\right) d\mathbf{g}, \tag{14}$$

where Δ represents the whole gradient domain, and we adopt the *Frobenius norm* to measure the difference between metric spaces. By taking derivative of the objective in (14) with respect to $\mathbf{G}$ and setting it to zero, we obtain

$$\mathbf{G}^\star = \iint_\Delta \mathbf{G}_0(\mathbf{g}) \cdot Pr\left(\mathbf{g} \,\middle|\, \{\mathbf{g}_k\}_{k=0}^{K-1}\right) d\mathbf{g}, \tag{15}$$

which integrates $\mathbf{G}_0(\mathbf{g})$ over the gradient domain. Note that integration of a matrix means integrate it component-wise. Moreover, by Bayes rules we can replace the posterior probability with the product of likelihood and prior probability:

$$\begin{aligned} Pr\left(\mathbf{g} \,\middle|\, \{\mathbf{g}_k\}_{k=0}^{K-1}\right) &\propto Pr(\mathbf{g}) \cdot Pr\left(\{\mathbf{g}_k\}_{k=0}^{K-1} \,\middle|\, \mathbf{g}\right) \\ &\propto Pr(\mathbf{g}) \cdot \prod_{k=0}^{K-1} Pr\left(\mathbf{g}_k | \mathbf{g}\right) \\ \propto \exp\left(-\frac{1}{2\sigma_g^2} \|\mathbf{g}\|_2^2\right) &\cdot \exp\left(-\frac{1}{2\sigma_e^2} \sum_{k=0}^{K-1} \|\mathbf{g} - \mathbf{g}_k\|_2^2\right), \end{aligned} \tag{16}$$

where we apply (13) and assume the prior $Pr(\mathbf{g})$ follows a 2D zero-mean Gaussian with covariance $\sigma_g^2 \mathbf{I}$. With further algebraic manipulation, one can show that the right-hand side of (16) is also a 2D Gaussian. Hence

$$Pr\left(\mathbf{g} \,\middle|\, \{\mathbf{g}_k\}_{k=0}^{K-1}\right) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{1}{2\sigma^2} \|\mathbf{g} - \mathbf{g}_\mu\|_2^2\right), \tag{17}$$

where its mean is $\mathbf{g}_\mu$ and covariance is $\sigma^2 \mathbf{I}$, expressed as:

$$\mathbf{g}_\mu = \frac{1}{K + \sigma_e^2/\sigma_g^2} \sum_{k=0}^{K-1} \mathbf{g}_k, \quad \sigma^2 = \frac{\sigma_e^2}{K + \sigma_e^2/\sigma_g^2}. \tag{18}$$

Note that σ^2 is a constant for different pixel locations on $\mathbf{p}_0$. Given (10) and (17), the integral of (15) has a closed-form expression

$$\mathbf{G}^\star = \mathbf{g}_\mu \mathbf{g}_\mu^{\mathrm{T}} + (\sigma^2 + \alpha)\mathbf{I}. \tag{19}$$

(19) has an intuitive interpretation: when the noise variance σ^2 of the patch is small, the first term dominates and the metric space is skewed and discriminant. When σ^2 is large — *i.e.*, when the estimated gradients $\mathbf{g}_k$ are unreliable, the second term dominates and the metric space is not skewed and is essentially the Euclidean space.

3.3. From Metric Space to Graph Laplacian

From Section 2.1, when operating on discrete images, we need discrete feature functions $\{\mathbf{f}_n^D\}_{n=1}^N$ to compute the graph weights, which subsequently lead to graph Laplacian $\mathbf{L}$. The simplicity of (19) allows us to select $N = 3$ appropriate feature functions by comparing (19) with (8), such that the selected feature functions will directly lead to the optimal metric space $\mathbf{G}^\star$ in (19). Let

$$f_1(x, y) = \sqrt{\sigma^2 + \alpha} \cdot x, \quad f_2(x, y) = \sqrt{\sigma^2 + \alpha} \cdot y. \tag{20}$$

[2]For simplicity, we assume the i.i.d. AWGN condition in the pixel domain carries to the gradient domain.

Table 1. Natural image denoising with OGLRD: performance comparisons in PSNR (dB) with NLM and BF

Image	Method	Standard Deviation σ_n				
		10	15	20	25	30
Lena	OGLRD	**35.12**	**33.53**	**32.33**	**31.38**	**30.64**
	NLM	34.26	32.03	31.51	30.38	29.45
	BF	29.48	27.00	24.80	23.00	21.52
Boats	OGLRD	**33.19**	**31.39**	**30.21**	**29.23**	**28.54**
	NLM	32.88	30.69	29.74	28.62	27.68
	BF	27.91	26.42	24.89	23.47	22.19
Pepp.	OGLRD	**34.70**	**33.31**	**32.26**	**31.51**	**30.81**
	NLM	33.97	31.96	31.48	30.42	29.50
	BF	28.96	26.70	24.67	22.95	21.49
Airpl.	OGLRD	**35.29**	**33.48**	**32.14**	**31.13**	**30.29**
	NLM	34.42	32.13	31.20	30.04	29.08
	BF	30.39	28.15	25.96	24.04	22.40

According to (8), $f_1(x, y)$ and $f_2(x, y)$ lead to the term $(\sigma^2 + \alpha)\mathbf{I}$ in (19). Correspondingly, in the discrete domain,

$$\mathbf{f}_1^D(i) = \sqrt{\sigma^2 + \alpha} \cdot x_i, \;\; \mathbf{f}_2^D(i) = \sqrt{\sigma^2 + \alpha} \cdot y_i. \qquad (21)$$

Recall that (x_i, y_i) denotes the coordinate of pixel i. And let

$$\mathbf{f}_3^D = \frac{1}{K + \sigma_e^2/\sigma_g^2} \sum_{k=0}^{K-1} \mathbf{p}_k, \qquad (22)$$

which sums the whole cluster $\{\mathbf{p}_k\}_{k=0}^{K-1}$. From the definition of $\mathbf{g}_\mu$ (18), $\mathbf{f}_3^D$ leads to the term $\mathbf{g}_\mu \mathbf{g}_\mu^{\mathrm{T}}$ in (19). With the defined $\mathbf{f}_1^D$, $\mathbf{f}_2^D$ and $\mathbf{f}_3^D$, we can obtain the neighborhood graph $\mathcal{G}$ and therefore the graph Laplacian $\mathbf{L}$ according to Section 2.1. Notice that from (21) and (22), our optimal graph Laplacian is closely-related to *joint* (or *cross*) bilateral filtering [15, 16] using the averaging of similar patches as guidance image. In contrast, our proposal adapts to the noise variance, resulting in more robust weight estimates.

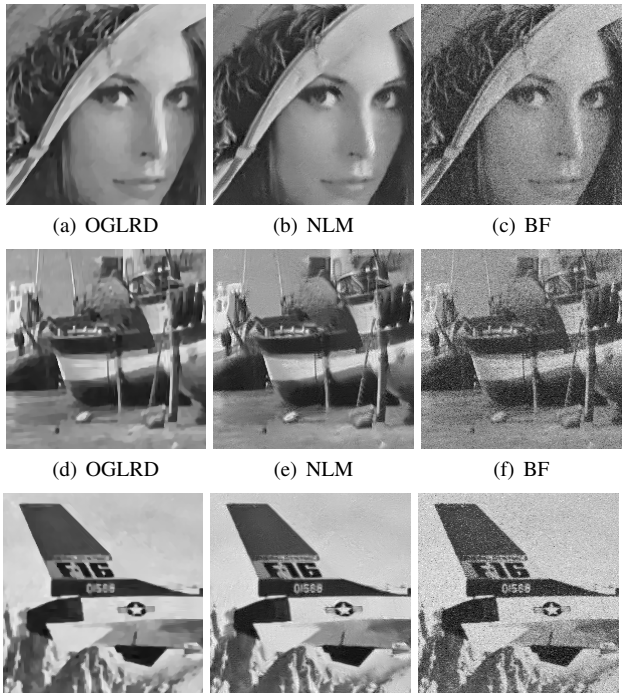
We estimate the constant σ_e^2 (variance of the noisy gradient) from the cluster $\{\mathbf{p}_k\}_{k=0}^{K-1}$. We first estimate the gradients of patches $\{\mathbf{p}_k\}_{k=0}^{K-1}$ with two filters $\mathbf{H}_x = [1 \; -1]$ and $\mathbf{H}_y = [1 \; -1]^{\mathrm{T}}$, leading to 2D gradients $\mathbf{g}_k^i = [g_{k,1}^i \; g_{k,2}^i]^{\mathrm{T}}$ for $0 \le k \le K-1$ and $1 \le i \le M$. Then at each pixel i, we compute the sample variances of $\{g_{k,1}^i\}_{k=0}^{K-1}$ and $\{g_{k,2}^i\}_{k=0}^{K-1}$ respectively. Then σ_e^2 is estimated as the mean of all the obtained sample variances.

3.4. Optimization Algorithm

We perform optimal local graph-based filtering for a noisy patch $\mathbf{p}_0$ as follows. We first search for its $K-1$ most similar patches via block-matching and Euclidean distance as metric. We then compute the feature functions, which lead to edge weights and graph Laplacian $\mathbf{L}$ as described in Section 3.3. Note that to reduce the effect of noise and obtain a good Laplacian $\mathbf{L}$, the aforementioned two steps are both performed on a bilateral-filtered version of the noisy image [9]. Having computed graph Laplacian $\mathbf{L}$, we can compute the optimally filtered output using (2). We denoise the given image iteratively to gradually enhance the image quality. At the end of the i-th iteration, the denoised patches are aggregated to form the updated denoised image. Our method has the same time complexity as NLM due to the block matching process. We name our method *optimal graph Laplacian regularization for denoising* (OGLRD).

4. EXPERIMENTATION

In this section, we evaluate the denoising performance of the proposed OGLRD and compare it against competing natural image denoising algorithms. Four 512×512 gray-scale images: *Lena*, *Boats*, *Peppers* and *Airplane* was used as the test images. The images were corrupted by i.i.d. AWGN, with standard deviation σ_n ranging from 10 to 30. We compared OGLRD with two competing methods: bilateral filtering (BF) [9] and non-local means (NLM) [8].



(a) OGLRD (b) NLM (c) BF

(d) OGLRD (e) NLM (f) BF

(g) OGLRD (h) NLM (i) BF

Fig. 3. Fragments of different denoised versions of the images *Lena*, *Boats*, and *Airplane*. The original images are corrupted by AWGN with $\sigma_n = 25$.

In our implementation, the patch size $\sqrt{M}$ was set as 25. We set the threshold r in (6) such that each vertex of $\mathcal{G}$ had at least eight edges, the normalization factor γ in (5) was chosen as 0.4. We let σ_g be a large value 10^6, while the constant α be a small value 10^{-12}. We run 3 iterations to denoise the images for a reasonable tradeoff between complexity and denoising quality.

Objective performance (measured in PSNR) of the proposed method, accompanied with those of the competing methods, are tabulated in Table 1. We see that OLGRD provided superior denoising performance, achieving up to 1.5 dB gain over NLM (*Lena*, $\sigma_n = 15$) and 9.3 dB gain over BF (*Peppers*, $\sigma_n = 30$).

Visual comparisons of OGLRD to NLM and BF are shown in Fig. 3, which shows segments from different denoised versions of $Lena$, $Boats$ and $Airplane$ corrupted by AWGN ($\sigma_n = 25$). We see that results of OGLRD exhibit natural and clear appearances, while NLM smears the details and BF fails to clean up the noise.

5. CONCLUSION

A key to graph-based local filtering for image denoising is the appropriate selection of edge weights in the graph that represents pixels in a target patch. In this paper, assuming patch gradients recur throughout an image due to self-similarity, we describe a methodology to derive the optimal edge weights given K nonlocal noisy observations of the target patch gradient. The resulting graph Laplacian regularizer is discriminant when the gradient estimates are accurate, and robust when the estimates are not. Experimental results show up to 1.5 dB denoising performance gain over non-local means (NLM).

6. REFERENCES

[1] P. Milanfar, "A tour of modern image filtering," *IEEE Signal Processing Magazine*, vol. 30, no. 1, pp. 106–128, January 2013.

[2] Leonid I Rudin, Stanley Osher, and Emad Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D: Nonlinear Phenomena*, vol. 60, no. 1, pp. 259–268, 1992.

[3] X. Wu and X. Zhang, "Model-based non-linear estimation for adaptive image restoration," in *IEEE Int'l Conf. Acoustics, Speech and Signal Processing*, April 2009, pp. 1185–1188.

[4] A. Kheradmand and P. Milanfar, "A general framework for kernel similarity-based image denoising," in *IEEE Global Conf. Sig. Info. Process.*, 2013, pp. 415–418.

[5] X. Liu, D. Zhai, D. Zhao, G. Zhai, and W. Gao, "Progressive image denoising through hybrid graph Laplacian regularization: a unified framework," *IEEE Trans. Imag. Process.*, vol. 23, no. 4, pp. 1491–1503, 2014.

[6] P. Wan, G. Cheung, D. Florencio, C. Zhang, and O.C. Au, "Image bit-depth enhancement via maximum-a-posterior estimation of graph AC component," in *IEEE Int'l Conf. Imag. Process.*, 2014, to appear.

[7] D.I. Shuman, S.K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Processing Magazine*, vol. 30, no. 3, pp. 83–98, 2013.

[8] A. Buades, B. Coll, and J-M Morel, "A non-local algorithm for image denoising," in *IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2005, vol. 2, pp. 60–65.

[9] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *IEEE Int'l Conf. Computer Vision*, 1998, pp. 839–846.

[10] H. G. Barrow and J. M. Tenenbaum, "Recovering intrinsic scene characteristics from images," *Computer Vision Systems*, pp. 3–26, 1978.

[11] W. Hu, X. Li, G. Cheung, and O.C. Au, "Depth map denoising using graph-based transform and group sparsity," in *IEEE Int'l Workshop on Multimedia Siging Processing*, 2013.

[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*, 2014, to appear.

[13] Hans Knutsson, Carl-Fredrik Westin, and Mats Andersson, "Representing local structure using tensors II," in *Image Analysis*, vol. 6688, pp. 545–556. Springer, 2011.

[14] M. Hein, "Uniform convergence of adaptive graph-based regularization," in *Learning Theory*, pp. 50–64. Springer, 2006.

[15] E. Eisemann and F. Durand, "Flash photography enhancement via intrinsic relighting," in *ACM SIGGRAPH*, 2004, pp. 673–678.

[16] G. Petschnigg, R. Szeliski, M. Agrawala, M. Cohen, H. Hoppe, and K. Toyama, "Digital photography with flash and no-flash image pairs," in *ACM SIGGRAPH*, 2004, pp. 664–672.