DOI: 10.19884/j.1672-5220.202409003

Graph-Based Transform and Dual Graph Laplacian Regularization for Depth Map Denoising

MENG Yaqun, GE Huayong*, HOU Xinxin, JI Yukai, LI Sisi
College of Information Science and Technology, Donghua University, Shanghai 201620, China

Abstract: Owing to the constraints of depth sensing technology, images acquired by depth cameras are inevitably mixed with various noises. For depth maps presented in gray values, this research proposes a novel denoising model, termed graph-based transform (GBT) and dual graph Laplacian regularization (DGLR) (DGLR-GBT). This model specifically aims to remove Gaussian white noise by capitalizing on the nonlocal self-similarity (NSS) and the piecewise smoothness properties intrinsic to depth maps. Within the group sparse coding (GSC) framework, a combination of GBT and DGLR is implemented. Firstly, within each group, the graph is constructed by using estimates of the true values of the averaged blocks instead of the observations. Secondly, the graph Laplacian regular terms are constructed based on rows and columns of similar block groups, respectively. Lastly, the solution is obtained effectively by combining the alternating direction multiplication method (ADMM) with the weighted thresholding method within the domain of GBT.

Keywords: depth map; graph signal processing; dual graph Laplacian regularization (DGLR); graph-based transform (GBT); group sparse coding (GSC)
CLC number: TN911.73 **Document code**: A
Article ID: 1672-5220(2025)05-0534-09

Open Science Identity (OSID)

0 Introduction

The depth camera captures information regarding the position of an object in three-dimensional (3D) space, where each pixel value indicates the distance from the corresponding object to the camera. Depth maps captured with active sensors such as time-of-flight (ToF)-based cameras and structure light (SL)-based cameras suffer from various types of intrinsic artifacts[1]. There is some recent work on denoising of depth maps, where the availability of both color and texture images is assumed, and their correlations are exploited for denoising[2-8]. We do not assume the availability of color images in this work. As a practical matter, it is difficult to obtain high-quality color images in challenging environments with unreliable lighting conditions, such as darkrooms. Denoising methods for depth maps presented in gray values are analogous to those for gray-scale images. Pioneering researchers tended to adopt a filter-based method[9-11] to filter depth maps. The biggest advantage of this type of method is that it is fast and efficient. However, its disadvantage is quite obvious: the denoising effect is not satisfactory. Based on the Bayesian assumption, the problem of solving the original image $\boldsymbol{X}$ according to the maximum a posteriori probability can be formulated as

$$\boldsymbol{X} = \arg\min_{\boldsymbol{X}} \| \boldsymbol{Y} - \boldsymbol{X} \|_2^2 + \psi\varphi(\boldsymbol{X}), \quad (1)$$

where $\boldsymbol{Y}$ represents the noisy image; $\varphi(\boldsymbol{X})$ represents the prior term; ψ represents the parameter. Compared with the filter-based method, the prior-based image denoising method[12-21] possesses higher accuracy and robustness. A suitable prior is indeed crucial in image denoising tasks. Among the currently available best models, the low-rank (LR) based models have shown good performance in image denoising. For instance, the weighted nuclear norm minimization (WNNM) model[12] is a classical one leveraging the LR prior. The rank residual constraint (RRC) model[13] is another example. It reconceptualizes the rank minimization problem in terms of matrix approximation. Specifically, it tries to progressively approximate the underlying LR matrix by minimizing the rank residuals. The LR regularized group sparse coding (LR-GSC) model[14] also plays a significant role. It is proposed by exploiting the nonlocal self-similarity (NSS) of natural images. This model bridges the gap between group sparse coding (GSC) and joint sparsity (JS). It takes advantage of both the sparsity and the LR property of the dictionary domain coefficients of each group of similar patches. Likewise, the low-rankness guided group sparse representation (LGSR) model[15], which also utilizes the NSS of natural images, jointly exploits the sparsity and the LR prior knowledge of each group of similar patches within a unified framework. This further enriches the ways to use such priors for improving

Received date: 2024-09-10
Foundation item: National Natural Science Foundation of China (No. 62372100)
* Correspondence should be addressed to GE Huayong, email: gehuayong@dhu.edu.cn
Citation: MENG Y Q, GE H Y, HOU X X, et al. Graph-based transform and dual graph Laplacian regularization for depth map denoising[J]. *Journal of Donghua University (English Edition)*, 2025, 42(5): 534-542.

denoising results. However, these LR-based models have a common drawback. They are poor at restoring texture details. The main reason is that they focus primarily on minimizing the nuclear norm to capture the LR nature of the data. In this process, they often overlook the fine-grained information related to texture details. As a result, some crucial high-frequency components essential for representing texture details may be inadvertently discarded or weakened.

When these LR-based models are applied to depth maps, the problem remains. For example, when we try to enhance a depth map of an indoor scene captured by a 3D sensor, these models can effectively reduce noise in the depth data. But they fail to accurately restore the fine details like the edges of furniture or the textures of the walls. This is due to their inherent focus on the LR nature of the data, causing them to sacrifice the high-frequency components crucial for depicting these texture details.

In contrast, there are models that specialize in depth map processing, and these models are excellent at preserving texture details. The nonlocal graph-based transform (NLGBT) model[16] uses graph-based signal processing techniques. It constructs graph structures based on image pixels and utilizes the relationships between pixels to better capture texture details. The group-based nuclear norm and learning graph (GNNLG) model[17] also has its unique features. It integrates graph signals with the LR property. Specifically, it embeds the graph topology by utilizing streaming Laplacian matrices and then combines it with the kernel norm. However, both models are more limited in terms of the amount of available data and the noise level in the images. They perform poorly on depth maps rich in more texture details with higher levels of noise.

Inspired by the above models, we propose to combine graph-based transform (GBT) with dual graph Laplacian regularization (DGLR) for eliminating high-frequency components while preserving texture details in depth map denoising tasks. The main reason for this combination lies in the unique properties of GBT and DGLR. The GBT provides a flexible framework for manipulating the image data while maintaining its structural information. By transforming the image into a graph domain, we can leverage the relationships between pixels (represented as nodes in the graph) to better understand and handle the image's texture details. When combined with the DGLR, it allows us to selectively suppress the identified high-frequency components that are likely to be noise, while safeguarding the texture details that are crucial for the visual quality and semantic understanding of the depth map. Specifically, within each group, the graph is constructed by using estimates of the true values of the averaged blocks instead of the observations. Next, in particular, the graph Laplacian regularity terms are constructed based on rows and columns of similar block groups, respectively. Finally, a combination of the alternating direction multiplication method (ADMM) and the weighted thresholding method within the domain of GBT is used to attain a rapid and effective solution.

1 Proposed DGLR-GBT Model

1.1 Graph spectra and graph-based dictionaries

Graph signal processing generalizes the signal processing task to signals living on a non-Euclidean domain, whose structure can be captured by a weighted graph[22]. We define a weighted undirected graph $\boldsymbol{g} = (V, E, \boldsymbol{W})$, where V, E and $\boldsymbol{W}$ represent the vertex set, edge set and weight matrix, respectively. For an image, blocks of pixels in the image are generally considered as the vertices of the graph, and the weights of the weight matrix $\boldsymbol{W}$ are often computed with a Gaussian kernel. The degree matrix $\boldsymbol{D} = \mathrm{diag}(d_1, d_2, \cdots, d_n)$, where d_i is the sum of elements in the ith row of the weight matrix $\boldsymbol{W}$. Here $d_i = \sum_j w_{ij}$, where w_{ij} represents the element in the ith row and jth column of the weight matrix $\boldsymbol{W}$. The graph Laplacian matrix $\boldsymbol{L}$ is defined as $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W}$. Specifically, $\boldsymbol{L}$ can be decomposed as

$$\boldsymbol{L} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^{\mathrm{T}}, \tag{2}$$

where $\boldsymbol{U}$ is the matrix consisting of the eigenvectors; $\boldsymbol{\Lambda}$ is the diagonal matrix of eigenvalues. For a given signal $\boldsymbol{s}$, the graph signal smoothness is defined as $\boldsymbol{s}^{\mathrm{T}}\boldsymbol{L}\boldsymbol{s}$.

The graph Fourier transform (GFT) of $\boldsymbol{s}$ is $\boldsymbol{S} = \boldsymbol{U}^{\mathrm{T}}\boldsymbol{s}$, where $\boldsymbol{S}$ is the frequency component of $\boldsymbol{s}$ in the spectral domain. Obviously, the corresponding inverse GFT (IGFT) can also be obtained, which is expressed as $\boldsymbol{s} = \boldsymbol{U}\boldsymbol{S}$. From another point of view, $\boldsymbol{U}$ can be viewed as a dictionary, called a graph-based dictionary. For natural images, it is observed that the energy of the frequency components of the graph signal is concentrated in the low-frequency components. This characteristic enables us to utilize the low-frequency components for the sparse representation of the graph signal. Through the above analysis, it becomes clear that the GBT can be effectively combined with the sparse representation optimization problem. When performing graph dictionary construction, the element w_{ij} of the weight matrix $\boldsymbol{W}$ is defined as

$$w_{ij} = \exp(\mathrm{dist}(i, j)/\sigma_w^2), \tag{3}$$

where σ_w^2 is the Gaussian kernel; $\mathrm{dist}(i, j)$ represents a distance between the node i and the node j. In the context of distance metric application, the general distance metric often utilizes its observed value $\boldsymbol{y}_i$. For instance, it defines the distance between two points i and j as $\mathrm{dist}(\hat{i}, j) = \sum_{k=1}^{K} [\boldsymbol{y}_k(i) - \boldsymbol{y}_k(j)]^2/K$, where K is the number of nearest neighbors. However, this approach ignores the effect of noise. When the noise level is relatively high, the matching accuracy based on this distance metric will be significantly reduced. To address

this issue, we propose to use the modified true value $\boldsymbol{x}_i$ instead of $\boldsymbol{y}_i$ to redefine the distance metric. Specifically, $\mathrm{dist}(i, j) = \sum_{k=1}^{K} [\boldsymbol{x}_k(i) - \boldsymbol{x}_k(j)]^2 / K$. We consider the difference $\{\boldsymbol{y}_k(i) - \boldsymbol{y}_k(j)\}_{k=1}^{K}$ to be the K sample values of the random variable $\{\boldsymbol{x}_k(i) - \boldsymbol{x}_k(j)\}_{k=1}^{K}$. According to the law of large numbers, we can establish the relationship that $\{\boldsymbol{y}_k(i) - \boldsymbol{y}_k(j)\}_{k=1}^{K} = \{\boldsymbol{x}_k(i) - \boldsymbol{x}_k(j)\}_{k=1}^{K} + e$. Here, the random variable e obeys a Gaussian distribution with a mean of 0. We use $E[\boldsymbol{y}_k(i) - \boldsymbol{y}_k(j)] - e$ in place of $E[\boldsymbol{x}_k(i) - \boldsymbol{x}_k(j)]$ and $D[\boldsymbol{y}_k(i) - \boldsymbol{y}_k(j)]$ in place of $D[\boldsymbol{x}_k(i) - \boldsymbol{x}_k(j)]$. So, $\mathrm{dist}(i, j) = \frac{1}{K}\sum_{k=1}^{K} [\boldsymbol{x}_k(i) - \boldsymbol{x}_k(j)]^2 = E[(\boldsymbol{x}_k(i) - \boldsymbol{x}_k(j))^2] = E^2[\boldsymbol{x}_k(i) - \boldsymbol{x}_k(j)] + D[\boldsymbol{x}_k(i) - \boldsymbol{x}_k(j)] \approx E^2[\boldsymbol{y}_k(i) - \boldsymbol{y}_k(j)] + e^2 + D[\boldsymbol{y}_k(i) - \boldsymbol{y}_k(j)] = \mathrm{dist}(\hat{i}, j) + e^2$.

The GBT-specific build process[16] proceeds as follows. Firstly, similar blocks are grouped and averaged to get an average block. This averaging process serves multiple purposes: 1) it smooths out small variations or noise within the individual similar blocks; 2) it yields a stable and representative description of the common features present in those blocks; 3) it provides a single entity that encapsulates the key characteristics of the grouped similar regions. Next, the pixels of this average block as the nodes of the graph and connected to the four nearest neighbors in the patch to form a four-connected graph. Given the connectivity graph, we can compute the adjacency matrix $\boldsymbol{A}$ and the degree matrix $\boldsymbol{D}$. The adjacency matrix $\boldsymbol{A}$ is a fundamental representation of the graph's structure. It records the presence or absence of edges between nodes. In our four-connected graph context, if there is an edge connecting two pixels (nodes), the corresponding entry in the adjacency matrix $\boldsymbol{A}$ will be 1, and 0 otherwise. The weight matrix $\boldsymbol{W}$ of the constructed graph is measured by the square of the intensity difference between neighboring pixels. Subsequently, we can obtain the Laplacian matrix $\boldsymbol{L}$. The eigenvectors of the Laplacian matrix $\boldsymbol{L}$ carry valuable information about the frequency components and spatial patterns within the graph structure, which forms the basis for constructing the GBT dictionary. Finally, we use the eigenvectors of the Laplacian matrix $\boldsymbol{L}$ to construct the GBT dictionary $\boldsymbol{U}$. By using these eigenvectors as the basis vectors for the dictionary, we can represent the image data in a transformed domain that captures both the local and global characteristics of the image.

1.2 DGLR

An image block group is the basic processing unit for image denoising. As shown in Fig. 1, given a clean image $\boldsymbol{x}$, we divide it into J overlapping image patches $\{\boldsymbol{x}_i\}_{i=1}^{i=J}$ of size $n \times n$. For each image patch, the K nearest neighbor method is used to find its most similar $K-1$ nearest neighbors to get the similar patches $\boldsymbol{x}_i^K$, where the similarity between two image blocks is measured by the Euclidean distance. Since each image block can be expanded into an $\mathbf{R}^{n\times 1}$ column vector, then each similar block group can be denoted as $\boldsymbol{G}_i^{r\times c} \in \mathbf{R}^{n\times K}$, where r and c represent the row and column indices of the matrix consisting of the similar block group, respectively. We normalize the K similar blocks, i.e., the rows of the group $\boldsymbol{G}_i^{r\times c}$, to obtain J normalized blocks denoted as $\boldsymbol{x}_i$.

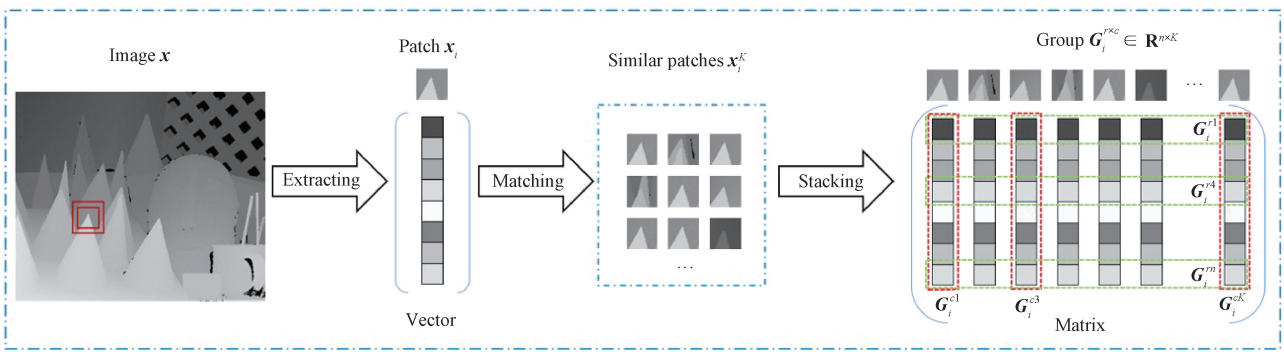

Fig. 1 Schematic diagram of group acquisition

A feature of the above group is that similar patches are usually sparse. Thus, $\boldsymbol{x}$ can be expressed as $\boldsymbol{x} = \boldsymbol{U}\boldsymbol{\alpha}$, where $\boldsymbol{\alpha}$ denotes the correlation coefficient. When we get the dictionary $\boldsymbol{U}$, the sparse coding can be expressed as

$$\min_{\boldsymbol{\alpha}} \| \boldsymbol{y} - \boldsymbol{x} \|_2^2 + \theta \| \boldsymbol{\alpha} \|_0 \quad \text{s.t.} \quad \boldsymbol{x} = \boldsymbol{U}\boldsymbol{\alpha}, \tag{4}$$

where $\boldsymbol{y}$ represents the noisy image; θ represents the regularization parameter. After solving the sparse coding problem, we can get the noise reduction group $\boldsymbol{x} = \boldsymbol{U}\boldsymbol{\alpha}$ from the obtained $\boldsymbol{\alpha}$.

For simplicity, we use the subscript r and subscript c to represent rows and columns, respectively. We regard the rows of the similar block group $\boldsymbol{G}_i^{r\times c}$ as the vertices $\boldsymbol{V}_r$ of the graph $\boldsymbol{g}_r = (\boldsymbol{V}_r, \boldsymbol{E}_r, \boldsymbol{W}_r)$, where $\boldsymbol{E}_r \subset \boldsymbol{V}_r \times \boldsymbol{V}_r$, and the weights $\boldsymbol{W}_r$ are solved by Eq. (3). Next, we can get the row graph Laplacian matrix $\boldsymbol{L}_r = \boldsymbol{D}_r - \boldsymbol{W}_r$, where $\boldsymbol{D}_r$ is the row degree matrix. Based on this, the row graph Laplacian regular term $\mathrm{tr}(\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{L}_r\boldsymbol{\alpha})$ can be obtained from Eq. (2). In the same way, we can also obtain the column graph Laplacian regular term $\mathrm{tr}(\boldsymbol{\alpha}\boldsymbol{L}_c\boldsymbol{\alpha}^{\mathrm{T}})$.

2 Proposed Solution

Group-based methods have received much attention in image denoising. However, the existing methods, with a predominant focus on sparsity or LR property, might impose limitations on the performance of image denoising. In this research, we propose a model integrating GBT and DGLR under sparsity constraints, which is denoted as DGLR-GBT. Figure 2 illustrates the framework structure of the proposed model DGLR-GBT. Here t_w is a new weighted threshold; μ, λ and θ are regularization parameters.

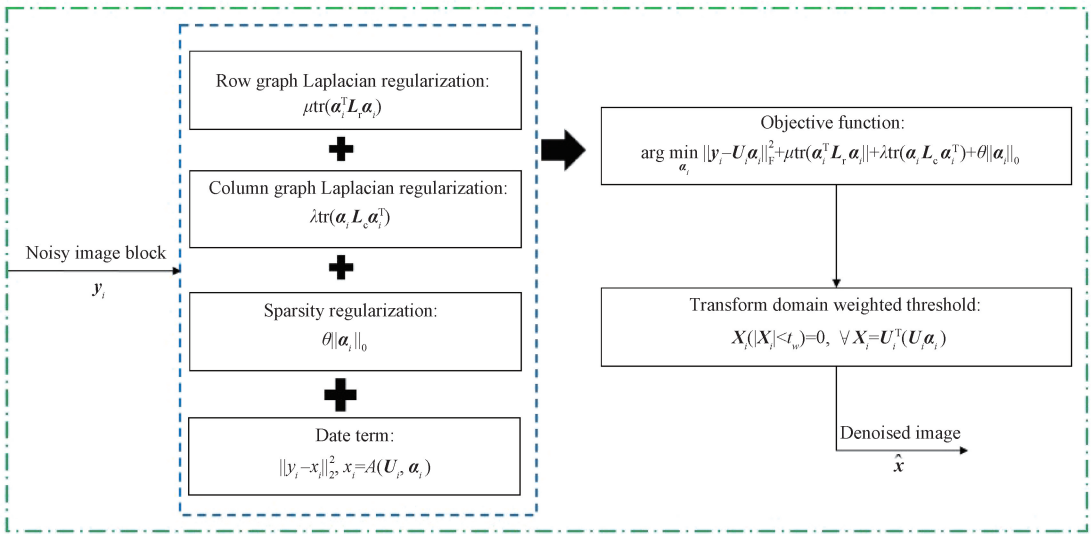

Fig. 2 Framework structure of DGLR-GBT

2.1 Dual graph regularized sparse coding

A single prior is difficult to effectively utilize all the properties of depth maps, and there are local differences in the statistical properties of the noise in the matrix. Therefore, in this research, we introduce the row-graph Laplacian regularization term to enhance the local smoothness of the depth map. Meanwhile, we also introduce the column-graph Laplacian regularization term to make full use of the NSS prior to the natural image. Since two Laplacian regularization terms are added to GBT, it is named DGLR-GBT and can be composed as

$$\arg\min_{\boldsymbol{\alpha}_i}\|\boldsymbol{y}_i-\boldsymbol{U}_i\boldsymbol{\alpha}_i\|_{\mathrm{F}}^2+\mu\mathrm{tr}(\boldsymbol{\alpha}_i^{\mathrm{T}}\boldsymbol{L}_{\mathrm{r}}\boldsymbol{\alpha}_i)+\lambda\mathrm{tr}(\boldsymbol{\alpha}_i\boldsymbol{L}_{\mathrm{c}}\boldsymbol{\alpha}_i^{\mathrm{T}})+\theta\|\boldsymbol{\alpha}_i\|_0. \quad (5)$$

With $\boldsymbol{\alpha}_i$, the reconstructed image can be expressed by $\boldsymbol{x}_i=\boldsymbol{U}_i\boldsymbol{\alpha}_i$. We provide an algorithm based on ADMM, enabling the simultaneous update of all columns of $\boldsymbol{\alpha}_i$. For simplicity and without loss of generality, the subscript i is omitted in the derivation. The denoising Problem (5) is reformulated as

$$\arg\min_{\boldsymbol{\alpha}}\|\boldsymbol{y}-\boldsymbol{U}\boldsymbol{\alpha}\|_{\mathrm{F}}^2+\mu\mathrm{tr}(\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{L}_{\mathrm{r}}\boldsymbol{\alpha})+\lambda\mathrm{tr}(\boldsymbol{\alpha}\boldsymbol{L}_{\mathrm{c}}\boldsymbol{\alpha}^{\mathrm{T}})\quad \text{s.t. } \boldsymbol{\alpha}=\boldsymbol{Z}\ \|z_j\|_0\leqslant T\forall j. \quad (6)$$

The augmented Lagrangian function of Problem (6) is $L_\rho(\boldsymbol{\alpha},\boldsymbol{Z},\boldsymbol{V})=F(\boldsymbol{\alpha})+I(\boldsymbol{Z})+\rho\|\boldsymbol{\alpha}-\boldsymbol{Z}+\boldsymbol{V}\|_2^2$. Here $\boldsymbol{Z}$ is the dyadic variable; $\boldsymbol{V}$ denotes the scaled dual form variable; $F(\boldsymbol{\alpha})=\|\boldsymbol{y}-\boldsymbol{Z}\boldsymbol{\alpha}\|_{\mathrm{F}}^2+\mu\mathrm{tr}(\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{L}_{\mathrm{r}}\boldsymbol{\alpha})+\lambda\mathrm{tr}(\boldsymbol{\alpha}\boldsymbol{L}_{\mathrm{c}}\boldsymbol{\alpha}^{\mathrm{T}})$, for an indicator function; $I(\boldsymbol{Z})=\mathrm{I}(\|z_j\|_0\leqslant T,\forall j)$ returns 1 if the condition is satisfied and 0 otherwise; T is the sparsity parameter, which limits the upper bound of the number of non-zero elements in each z_j; ρ is the augmented Lagrangian parameter. Specifically, by applying the ADMM scheme, Problem (6) can be converted into

$$\boldsymbol{\alpha}^{\xi+1}=\arg\min_{\boldsymbol{\alpha}}L_\rho(\boldsymbol{\alpha},\boldsymbol{Z}^{\xi},\boldsymbol{V}^{\xi}), \quad (7)$$

$$\boldsymbol{Z}^{\xi+1}=\arg\min_{\boldsymbol{Z}}L_\rho(\boldsymbol{\alpha}^{\xi+1},\boldsymbol{Z},\boldsymbol{V}^{\xi}), \quad (8)$$

$$\boldsymbol{V}^{\xi+1}=\boldsymbol{V}^{\xi}+\boldsymbol{\alpha}^{\xi+1}-\boldsymbol{Z}^{\xi+1}, \quad (9)$$

where ξ is the iteration number. Based on the ADMM algorithm, an efficient solution can be obtained for each subproblem.

2.2 Transform domain weighted threshold

Upon accomplishing the determination of the graph dictionary and resolving the sparse coding issue, we proceed to further threshold the denoising group. The traditional empirical threshold[16] is defined as $t=\sigma\sqrt{2\log(n^2K)}$, where σ represents the noise level, and it will be updated iteratively. While the traditional empirical threshold performs satisfactorily when the noise level is relatively low, it is prone to inducing excessive smoothing of the image in the presence of high noise levels, thereby impairing the edge information of the image. To address this issue, we introduce a weighting factor w_σ to the traditional empirical threshold t to alleviate the over-smoothing effect when the noise level is high.

This method can balance the effect of the noise level to some extent and improve the effectiveness of thresholding. The weighting factor w_σ is configured as

$$w_\sigma = \begin{cases} -0.01\sigma^{(0)} + 0.95 + 0.05\text{sign}(30 - \sigma^{(0)}), & \sigma^{(0)} \leqslant 50, \\ 0.4, & \sigma^{(0)} > 50, \end{cases} \tag{10}$$

where $\sigma^{(0)}$ denotes the initial value of the standard deviation of the noise. It is evident that the magnitude of the weights is adaptive to the noise level. Specifically, as the noise level escalates, a greater amount of interfering information affects the image edges. However, concomitantly, the more high-frequency values present in the frequency domain, the smaller the weights. The complete DGLR-GBT encoding is presented in Algorithm 1. The full code related to this study can be accessed at https://github.com/mengyaqun/DGLR-GBT.git.

Algorithm 1 Depth map denoising via DGLR-GBT

Input: Noisy image $\boldsymbol{y}$, Noise level σ

Initialization: $\boldsymbol{y}^{(1)} = \boldsymbol{y}$

for $i = 1$ to *iter* do

1. Block clustering, as shown in Fig. 1
2. GBT dictionary construction is carried out based on Eq. (2) to obtain $\boldsymbol{U}$
3. Graph Laplacian regular term construction: $\text{tr}(\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{L}_{\mathrm{r}}\boldsymbol{\alpha})$, $\text{tr}(\boldsymbol{\alpha}\boldsymbol{L}_{\mathrm{c}}\boldsymbol{\alpha}^{\mathrm{T}})$
4. Problem (6) is solved by the ADMM algorithm
5. Transform spectrum shrinkage: the new threshold $w_\sigma \times t$ according to Eq. (10)

Output: The denoised depth map $\hat{\boldsymbol{x}}$

3 Experimental Results

3.1 Experimental setup

For the assessment of the quality of denoised images, the peak signal-to-noise ratio (PSNR) and the structure similarity (SSIM) are employed as evaluation metrics. All experiments are conducted in the MATLAB R2022b environment and executed on an Intel(R) Core (TM) i5-10210U CPU @ 1.60 GHz 2.11 GHz processor. Subsequently, the parameter settings within our proposed method are deliberated. The regularization parameters μ, λ and θ are set as 1, 1 and 1, respectively, and the weighted threshold t_w is set as $w_\sigma \times t$. Other parameters, such as the block size n, the number of similar blocks K, and the relaxation parameter δ for iterative regularization with the noise variance updating parameter η are assigned values as presented in Table 1.

Table 1 Parameter settings

σ/dB	n	K	δ	η
$\sigma \leqslant 20$	5	25	0.13	0.63
$20 < \sigma \leqslant 30$	6	30	0.09	0.63
$30 < \sigma \leqslant 40$	7	50	0.08	0.65
$40 < \sigma \leqslant 50$	8	60	0.08	0.67
$50 < \sigma$	9	80	0.08	0.67

3.2 Objective evaluation

The widely used Middlebury stereo datasets[23-25] for depth map denoising are employed in our experiments. These datasets include the 2003, 2005 and 2006 versions. An additive Gaussian white noise with a standard deviation of 10–50 is superimposed onto these images. Several comparison algorithms are employed, including block-matching and 3D filtering algorithm (BM3D)[11], NLGBT[16], WNNM[12], RRC[13], LR-GSC[14], multi-scale weighted group sparse coding (MS-WGSC) model[17] and LGSR[15]. Tables 2 and 3 display the average values of PSNR and SSIM for all the depth maps within the Middlebury stereo datasets. The optimal and sub-optimal performances are indicated by red bold and green text, respectively.

As can be seen from Table 2, DGLR-GBT surpasses all the state-of-the-art algorithms in terms of the average PSNR. This encompasses even the powerful LGSR[15] and WNNM[12]. From Table 3, it is evident that both LGSR[15] and DGLR-GBT achieve the optimal results in nearly all the average values of SSIM for all the depth maps. It is worth highlighting that LGSR[15] represents the latest and most advanced denoising algorithm that combines the sparse and LR priors. However, it is conspicuous that the SSIM values of LGSR decline compared to those of the other models when the noise levels are high, particularly when the noise levels fall within a range of 40–50 dB. This is due to the fact that some pseudo-structures are introduced into the noisy image when the noise level is high, which diminishes the structural similarity of the image and consequently restricts the denoising performance.

To further validate the effectiveness of our algorithm, we conduct comparative experiments with two classic denoising algorithms: the feed-forward denoising convolutional neural network (DnCNN)[26], and the fast and flexible denoising convolutional neural network (FFDNet)[27]. Additionally, to verify the robustness of our algorithm, we evaluate it on a real-world depth dataset, RGBM. The results are presented in Tables 4 and 5, respectively, with the best values highlighted in bold. As shown in Table 4, DGLR-GBT outperforms DnCNN and achieves comparable performance to FFDNet in terms of denoising quality. Furthermore, Table 5 demonstrates that DGLR-GBT maintains a stable advantage over other depth map datasets.

Table 2 PSNR comparison of various algorithms on Middlebury stereo datasets

σ/dB	PSNR							
	BM3D	NLGBT	WNNM	RRC	LR-GSC	MS-WGSC	LGSR	DGLR-GBT
10	42.603 7	43.979 3	44.011 2	42.968 8	43.462 1	43.871 6	44.058 0	**44.514 6**
15	39.560 0	40.740 0	41.092 1	40.361 1	39.111 4	40.957 8	40.874 3	**41.399 9**
20	37.543 1	38.166 6	38.972 2	38.408 8	38.818 7	38.876 6	38.988 8	**39.401 1**
25	36.080 4	36.149 2	37.398 3	36.931 0	36.772 0	37.161 8	37.371 0	**37.585 0**
30	35.027 7	34.634 6	36.187 0	35.786 8	36.197 7	35.950 5	36.245 0	**36.263 7**
40	33.402 9	32.260 9	34.380 7	34.076 0	34.606 2	34.128 8	34.510 3	**34.631 6**
50	32.140 0	31.048 9	33.196 8	32.776 7	33.332 5	33.393 9	33.242 0	**33.427 1**
Average	36.622 5	36.711 4	37.891 2	37.329 9	37.471 5	37.763 0	37.898 5	**38.174 7**

Table 3 SSIM comparison of various algorithms on Middlebury stereo datasets

σ/dB	SSIM							
	BM3D	NLGBT	WNNM	RRC	LR-GSC	MS-WGSC	LGSR	DGLR-GBT
10	0.987 8	0.988 6	0.990 2	0.989 9	0.990 7	0.989 7	**0.9922**	0.991 0
15	0.979 6	0.980 4	0.982 6	0.984 1	0.968 5	0.981 9	**0.985 9**	0.983 7
20	0.970 4	0.967 6	0.973 5	0.977 2	0.978 3	0.972 3	**0.980 3**	0.979 1
25	0.961 0	0.950 6	0.967 1	0.972 9	0.965 6	0.963 9	**0.975 7**	**0.975 7**
30	0.950 8	0.936 1	0.957 7	0.966 9	0.969 1	0.953 5	**0.970 4**	0.969 4
40	0.930 9	0.935 1	0.937 5	0.955 6	0.960 0	0.930 9	0.957 8	**0.963 9**
50	0.925 6	0.923 4	0.934 4	0.943 8	0.950 9	0.951 2	0.949 4	**0.957 1**
Average	0.958 0	0.954 5	0.963 3	0.9701	0.969 0	0.963 3	0.973 1	**0.974 3**

Table 4 PSNR and SSIM comparison of DnCNN, FFDNet and DGLR-GBT on Middlebury stereo datasets

σ/dB	DnCNN		FFDNet		DGLR-GBT	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
10	43.114 6	0.988 6	43.689 0	**0.991 0**	**44.514 6**	**0.991 0**
20	38.785 9	0.974 1	39.312 0	**0.982 5**	**39.401 1**	0.979 1
30	36.214 3	0.958 1	**36.727 3**	**0.972 7**	36.263 7	0.969 4
40	34.640 2	0.948 6	**35.026 5**	0.953 5	34.631 6	**0.963 9**
50	33.469 0	0.936 2	**33.804 6**	0.953 5	33.427 1	**0.957 1**
Average	37.244 8	0.961 1	**37.711 9**	0.970 6	37.647 6	**0.972 1**

Table 5 PSNR and SSIM comparison of various algorithms on RGBM datasets

σ/dB	Metric	Algorithm							
		BM3D	NLGBT	WNNM	RRC	LR-GSC	MS-WGSC	LGSR	DGLR-GBT
30	PSNR	33.374 6	33.124 0	33.970 0	33.849 9	33.837 7	34.066 3	33.829 3	**34.245 1**
	SSIM	0.945 7	0.946 6	0.956 6	0.967 5	0.964 3	0.952 3	0.966 3	**0.971 2**
50	PSNR	30.164 7	28.913 6	30.825 2	30.743 6	30.760 3	31.249 0	30.677 2	**31.259 0**
	SSIM	0.920 3	0.936 4	0.932 8	0.946 0	0.943 7	0.949 3	0.942 3	**0.955 2**

3.3 Subjective evaluation

Figures 3, 4 and 5 present the visual comparison among the competing denoising methods on depth maps Art, Teddy and Rocks2 at noise levels of 15, 30 and 50 dB, respectively. As depicted in Fig. 3, when the noise level is relatively low, all denoising methods yield satisfactory recovery results. Nevertheless, as the noise level escalates, WNNM[12], MS-WGSC[17], LR-GSC[14] and LGSR[15] begin to generate some undesirable artifacts, while BM3D[11], NLGBT[16] and RRC[13] result in some over-smoothing outcomes, as demonstrated in Figs. 4 and 5. Only DGLR-GBT performs stably in denoising tasks at different noise levels. This stability can be attributed to two key aspects of its design. On the one hand, DGLR-GBT incorporates a dual graph Laplacian structure that enables it to effectively analyze the spectral characteristics of the image in both local and global aspects. By doing so, it can precisely distinguish between

noise-related high-frequency components and those crucial texture details, regardless of the noise level. On the other hand, the GBT method allows it to adaptively adjust the denoising process according to the specific characteristics of different images and noise levels, ensuring that it neither generates artifacts nor causes over-smoothing.

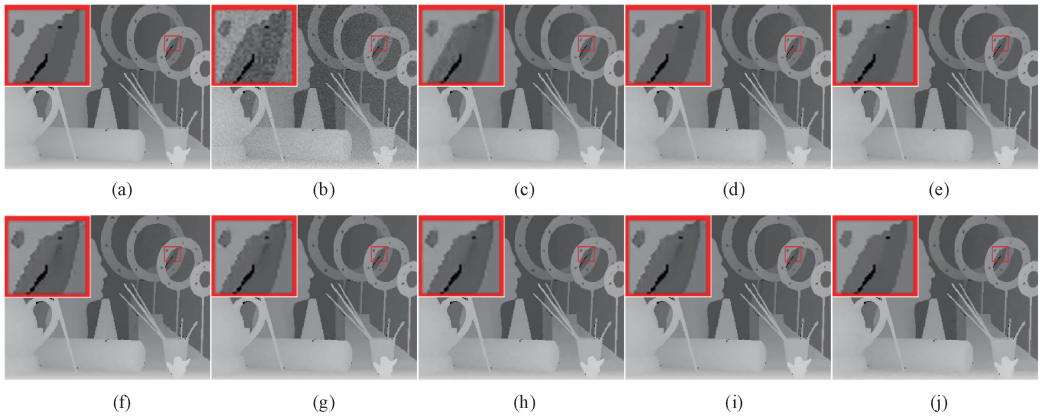

Fig. 3 Denoising results on depth map Art by different methods (noise level: 15 dB): (a) ground truth; (b) noisy image; (c) BM3D; (d) NLGBT; (e) WNNM; (f) RRC; (g) LR-GSC; (h) MS-WGSC; (i) LGSR; (j) DGLR-GBT

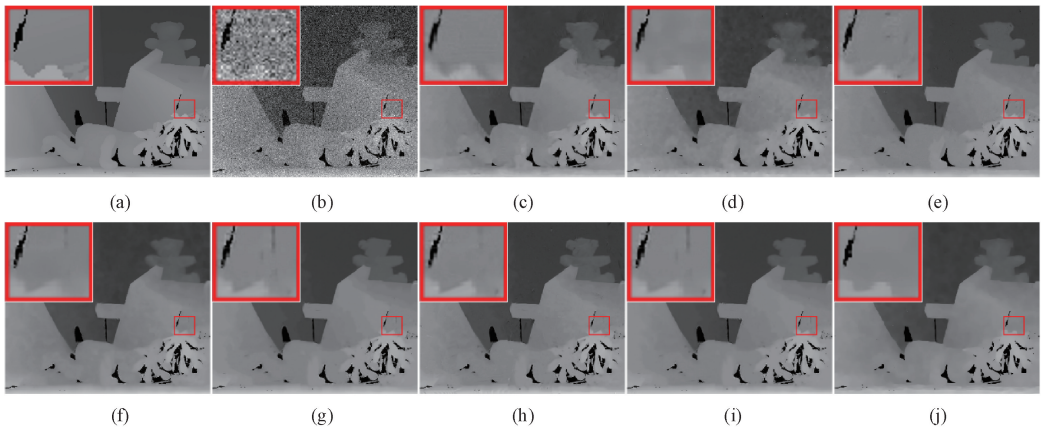

Fig. 4 Denoising results on depth map Teddy by different methods (noise level: 30 dB): (a) ground truth; (b) noisy image; (c) BM3D; (d) NLGBT; (e) WNNM; (f) RRC; (g) LR-GSC; (h) MS-WGSC; (i) LGSR; (j) DGLR-GBT

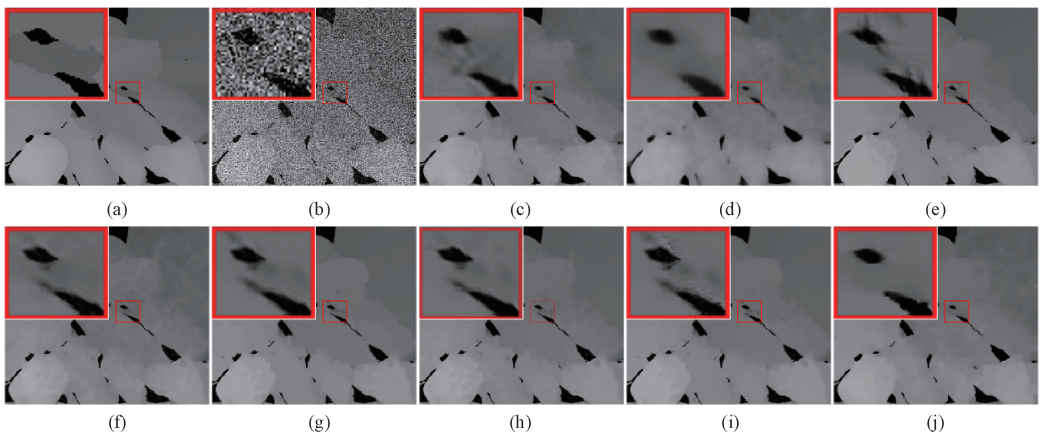

Fig. 5 Denoising results on depth map Rocks2 by different methods (noise level: 50 dB): (a) ground truth; (b) noisy image; (c) BM3D; (d) NLGBT; (e) WNNM; (f) RRC; (g) LR-GSC; (h) MS-WGSC; (i) LGSR; (j) DGLR-GBT

3.4 Algorithm complexity analysis

Considering the feasibility in practical applications, especially in real-time processing scenarios, a low computational cost is of crucial importance for depth map denoising. The complexity of the proposed algorithm is analyzed in the following manner. Firstly, an image with

a size of $N \times N$ is divided into overlapping image blocks with a size of $n \times n$. For each of these blocks, the construction of the graph and the solution process of the ADMM algorithm are the two main time-consuming operations. The time for these two operations are denoted by T_1 and T_2, respectively. Specifically, T_1 refers to the time required for generating the adjacency matrix $\boldsymbol{A}$, the degree matrix $\boldsymbol{D}$, and the Laplacian matrix $\boldsymbol{L}$. Through theoretical analysis and relevant calculation, it can be derived that $T_1 = O(n^2)$. As for T_2, it represents the time spent on solving the model presented in Problem (6). More precisely, T_2 can be decomposed into $t_1 + t_2$, where t_1 is the time taken for solving the $\boldsymbol{\alpha}$-subproblem, and t_2 is the time needed for solving the $\boldsymbol{Z}$-subproblem. Consequently, $T_2 = O(n^3)$. Based on the above analysis, the complexity of reconstructing the entire image can be expressed as $O(n^3)$. It is worth highlighting that parallel algorithms can be designed to reduce the elapsed time. Moreover, with the continuous development of hardware, particularly CPUs and GPUs, parallel computing can be readily implemented[8].

4 Conclusions

Group-based methodologies have attracted significant attention within the domain of image denoising. Nevertheless, the majority of the extant methods predominantly center around sparsity or LR property, which might potentially impede the performance of image denoising. In this research, we introduce a graph-based denoising framework that exhibits particular efficacy for depth maps. We propose a model that combines the graph dictionary and the dual graph Laplacian within the context of sparsity constraints. This model fully exploits NSS and the piecewise smoothness of the depth maps to eliminate noise. Concurrently, more comprehensive and detailed information is retained during the noise removal process via the combination of the ADMM and the GBT domain weighted thresholding.

References

[1] IBRAHIM M M, LIU Q, KHAN R, et al. Depth map artefacts reduction: a review[J]. *IET Image Processing*, 2020, 14(12): 2630-2644.

[2] ZHONG Z W, LIU X M, JIANG J J, et al. Guided depth map super-resolution: a survey[J]. *ACM Computing Surveys*, 2023, 55(14Sup.): 1-36.

[3] ZHANG Y B, FENG Y H, LIU X M, et al. Color-guided depth image recovery with adaptive data fidelity and transferred graph Laplacian regularization[J]. *IEEE Transactions on Circuits and Systems for Video Technology*, 2020, 30(2): 320-333.

[4] WANG Z X, YAN Z Q, YANG J. SGNet: structure guided network via gradient-frequency awareness for depth map super-resolution[J]. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024, 38(6): 5823-5831.

[5] ZHONG Z W, LIU X M, JIANG J J, et al. Deep attentional guided image filtering[J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2023, 35(9): 12236-12250.

[6] ZHAO Z X, ZHANG J S, XU S, et al. Discrete cosine transform network for guided depth map super-resolution [C]//2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). New York: IEEE, 2022: 5687-5697.

[7] WANG K, ZHAO L J, ZHANG J J, et al. Joint depth map super-resolution method via deep hybrid-cross guidance filter [J]. *Pattern Recognition*, 2023, 136: 109260.

[8] ZHANG Y Y, HE X H, CHEN H G, et al. Depth map super-resolution via learned nonlocal model and enhanced local regularization [J]. *Signal Processing*, 2024, 218: 109368.

[9] BUADES A, COLL B, MOREL J M. A non-local algorithm for image denoising[C]//2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05). New York: IEEE, 2005: 60-65.

[10] GASTAL E S L, OLIVEIRA M M. Adaptive manifolds for real-time high-dimensional filtering [J]. *ACM Transactions on Graphics*, 2012, 31(4): 1-13.

[11] DABOV K, FOI A, KATKOVNIK V, et al. Image denoising by sparse 3-D transform-domain collaborative filtering[J]. *IEEE Transactions on Image Processing*, 2007, 16(8): 2080-2095.

[12] GU S H, ZHANG L, ZUO W M, et al. Weighted nuclear norm minimization with application to image denoising[C]//2014 IEEE Conference on Computer Vision and Pattern Recognition. New York: IEEE, 2014: 2862-2869.

[13] ZHA Z Y, YUAN X, WEN B H, et al. From rank estimation to rank approximation: rank residual constraint for image restoration [J]. *IEEE Transactions on Image Processing*, 2019, 29: 3254-3269.

[14] ZHA Z Y, WEN B H, YUAN X, et al. Image restoration via reconciliation of group sparsity and low-rank models [J]. *IEEE Transactions on Image Processing*, 2021, 30: 5223-5238.

[15] ZHA Z Y, WEN B H, YUAN X, et al. Low-rankness guided group sparse representation for image restoration [J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2023, 34(10): 7593-7607.

[16] HU W, LI X, CHEUNG G, et al. Depth map denoising using graph-based transform and group sparsity [C]//2013 IEEE 15th International

Workshop on Multimedia Signal Processing (MMSP). New York: IEEE, 2013: 1-6.
[17] YAN C G, LI Z S, ZHANG Y B, et al. Depth image denoising using nuclear norm and learning graph model [J]. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 2020, 16(4): 1-17.
[18] OU Y, SWAMY M N S, LUO J Q, et al. Single image denoising via multi-scale weighted group sparse coding [J]. *Signal Processing*, 2022, 200: 108650.
[19] LI F, RU Y M, LV X G. Patch-based weighted SCAD prior for rician noise removal[J]. *Journal of Scientific Computing*, 2021, 90(1): 26.
[20] LIU S Q, HU Q, LI P F, et al. Speckle suppression based on weighted nuclear norm minimization and grey theory [J]. *IEEE Transactions on Geoscience and Remote Sensing*, 2019, 57(5): 2700-2708.
[21] LIU S Q, GAO L L, LEI Y, et al. SAR speckle removal using hybrid frequency modulations[J]. *IEEE Transactions on Geoscience and Remote Sensing*, 2021, 59(5): 3956-3966.
[22] MARTINS W A, LIMA J B, RICHARD C, et al. A primer on graph signal processing[M]// Signal Processing and Machine Learning Theory. Cambridge, Massachusetts: Academic Press, 2024: 961-1008.
[23] SCHARSTEIN D, SZELISKI R. High-accuracy stereo depth maps using structured light [C]// 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. New York: IEEE, 2003, 1: 195-202.
[24] SCHARSTEIN D, PAL C. Learning conditional random fields for stereo [C]//2007 IEEE Conference on Computer Vision and Pattern Recognition. New York: IEEE, 2007, 1-8: 1688.
[25] HIRSCHMULLER H, SCHARSTEIN D. Evaluation of cost functions for stereo matching [C]//2007 IEEE Conference on Computer Vision and Pattern Recognition. New York: IEEE, 2007, 1-8: 2134.
[26] ZHANG K, ZUO W M, CHEN Y J, et al. Beyond a Gaussian denoiser: residual learning of deep CNN for image denoising [J]. *IEEE Transactions on Image Processing*, 2017, 26(7): 3142-3155.
[27] ZHANG K, ZUO W M, ZHANG L. FFDNet: toward a fast and flexible solution for CNN based image denoising [J]. *IEEE Transactions on Image Processing*, 2018, 27(9): 4608-4622.

基于图变换和对偶图拉普拉斯正则化的深度图像去噪

孟亚群, 葛华勇*, 侯鑫鑫, 吉宇凯, 李思思
东华大学 信息科学与技术学院, 上海 201620

摘　要: 由于深度传感技术的限制, 深度相机获取的图像不可避免地会掺杂各种噪声。针对以灰度值呈现的深度图, 该研究提出了一种新的去噪模型, 称为基于图变换和对偶图拉普拉斯正则化。该模型利用深度图固有的非局部自相似性和分段平滑性等特性, 专门去除高斯白噪声。在组稀疏编码框架内, 实现了基于图变换和对偶图拉普拉斯正则化的结合。首先, 在每个组内, 使用平均块真实值的估计值而不是观测值来构建图。其次, 分别基于相似块组的行和列构建图拉普拉斯正则项。最后, 将交替方向乘法与图变换域内的加权阈值相结合来有效地求解模型。

关键词: 深度图; 图信号处理; 对偶图拉普拉斯正则化; 图变换; 组稀疏编码