We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,300

Open access books available

172,000

International authors and editors

190M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Image Restoration and Noise Reduction with Context-Dependent Wavelet Graph and ADMM Optimization

Jinn Ho, Shih-Shuo Tung and Wen-Liang Hwang

Abstract

We represent the image noise reduction and restoration problems as context-dependent graphs and propose algorithms to derive the optimal graphs by the alternating direction method of multipliers (ADMM) method. An image is spatially decomposed into smooth regions and singular regions, consisting of edges and textures. The graph representing a smooth region is defined in the image domain, while that representing a singular region is defined in the wavelet domain. The optimal graphs are formulated as the solutions of constrained optimization problems over sparse graphs, where the sparseness is imposed on the edges. The graphs on the wavelet domain are solved in a hierarchical layer structure. The convergence and complexity of the algorithms have been studied. Simulation experiments demonstrate that the results of our algorithms are superior to the state-of-the-art algorithms for image noise reduction and restoration.

Keywords: image restoration, image denoising, graph, ADMM, wavelet

1. Introduction

We consider the inverse problem of deriving the original image $x \in \mathcal{R}^N$ from an observation $y \in \mathcal{R}^N$, expressed as

$$y = Hx + n, \quad (1)$$

where H is an $N \times N$ matrix and $n \sim \mathcal{N}(0_{N \times 1}, \sigma I_{N \times N})$ is a vector of independent and identically distributed (i.i.d.) Gaussian random variables with standard deviation σ . We further assume that the point spread function H is known. By imposing prior information on the desired image, given as

$$\hat{x} = \arg \min_x \frac{1}{\sigma^2} \|y - Hx\|^2 + \text{penalty}(x), \quad (2)$$

where the first term is the data fidelity term for the Gaussian observation model and the second term is the regularization term, measuring the penalty of a solution that deviated away from the prior knowledge of the desired image. The modeling of the desired image is at the core of the approach [1–4]. The primary challenge of solving the problem is to recover the local high-frequency information of edges and texture in the original images that are not present in the observation.

If H is the identity matrix, problem (1) is called the noise reduction problem. The solutions vary with what type of noise is contaminated in the observation [5]. If the noise is white Gaussian noise, the state-of-the-art algorithms for the problem are BM3D [6], WBN [7], and DRUNet [8]. BM3D utilizes the tight frame representation of an image, where atoms of the frame are derived from image patches. WBN is a graphical probabilistic model of a weighted directed acyclic graph (DAG) in the wavelet domain. Different from BM3D and WBN, DRUNet is a deep learning method. It is a flexible and powerful deep CNN denoiser and the architecture is the combination of U-Net [9] and ResNet [10]. It not only outperforms the state-of-the-art deep Gaussian denoising models but also is suitable to solve plug-and-play image restoration.

If H is a blur singular matrix, problem (1) is called the image restoration problem. In the optimization-based method, the best image restoration performance both subjectively and objectively was derived from the algorithm IDD-BM3D [11]. It utilizes sparse synthetic and analytic models and de-couples the problem into blur inverse and noise reduction sub-problems, each of which is solved by a variational optimization approach. In deep learning, DPIR [8] replaces the denoising sub-problem of model-based optimization with a learning-based CNN denoiser prior which is DRUNet. By iteratively solving the data sub-problem and a prior sub-problem to restore the image.

In this chapter, we also present a restoration algorithm that combines the noise reduction algorithm with the proximal point method [12]. The primary technical contributions of our methods are the context-dependent graphical representations and the algorithms to derive the optimal graphs of each representation. Finding the optimal graph in a combinatorial way is extremely difficult and likely an NP-hard problem [13, 14]. Unlike the combinatorial approach, we impose constraints on edges and include edges in the optimal graph only when the constraints on the edges are active. This renders a computationally solvable optimization problem and the solution is a graph with only a small number of active edges.

Based on local content in an image, the context-dependent representation divides the image into singular and smooth areas. Singular areas, consisting of edges or texture, are represented and processed differently from the smooth areas. The graphs of singular areas are constructed based on the persistence and sparsity of wavelet coefficients of the image. The persistence is imposed on the inter-scale edges so that the solution at one scale can be used to confine that in adjacent scales. Meanwhile, the sparsity is imposed on the intra-scale edges that preserve the edges in which end nodes have similar intensity. In contrast, a graph of a smooth area is in the image domain and has only sparse intra-scale edges.

The algorithm to derive the optimal graphs, called graphical ADMM, is based on the alternating direction method of multipliers (ADMM) method [15, 16]. It is an efficient and robust algorithm since it breaks a complicated problem into smaller pieces, each of which is easier to handle. In our case, the node update is separated from the edge update in the optimization. In addition, for wavelet graphs, graphical ADMM approximates the multi-scale optimization problem into a sequence of sub-problems; each can be efficiently solved by convex optimization methods.

The chapter is organized as follows. In Section 2, we present the models and the construction for the context-dependent graphs. In Section 3, we formulate the noise reduction problem as a graph optimization model and present the graphical ADMM method to derive optimal graphs. In Section 4, the image restoration problem is formulated as the proximal point method that reduces the problem into a sequence of noise reduction problems, each being solved by the method in Section 3. In Section 5, experimental results and the principal differences between our and the compared methods are also discussed. Section 6 contains concluding marks.

2. Context-dependent graphical models

An image is comprised of features of edges, texture, and smooth areas. A common approach to obtain a good image processing result is to treat different features with different approaches [17, 18]. Following this approach, an image is partitioned into two types of blocks. A block containing an edge point or texture is a singular block, while the others are smooth blocks. To keep the flow, we delay the partitioning method of an image, which is described in part A of Section 5, but this is not necessary to accurately partition an image to achieve the performance demonstrated in this chapter. The singular and smooth blocks were handled with different graph optimization approaches: a singular block is in the wavelet domain, while a smooth block is in the image domain. In the wavelet domain, a singular block is represented by several weighted graphs, one corresponding to an orientation. If the wavelet transform has three orientations, LH, HL, and HH, then one graph is for LH sub-bands, another for HL sub-bands, and the third for HH sub-bands. The graph for one orientation is constructed as follows.

Each wavelet coefficient is associated with a node. Edges are comprised of inter-scale and intra-scale edges. An inter-scale edge connecting nodes in adjacent scales can direct either from a coarse scale to a finer scale or vice versa. The inter-scale edges are built-in and data-independent; they are constructed based on the wavelet persistence. In contrast, an intra-scale edge connecting nodes of the same scale is un-directed, data-dependent, and determined based on the sparseness from the graph optimization algorithm. Regularizations have been imposed on inter-scale edges to preserve the persistence of wavelet coefficients across scales and on intra-scale edges to preserve the similarity of wavelet coefficients on nodes at the two ends of an edge.

2.1 Inter-scale edges

Since wavelets can characterize singularities, representing singularities with wavelets can facilitate the restoration of edges and texture in an image. The persistence property of wavelets means that the wavelet coefficients dependency and correlations across scales. Thus, inter-scale edges were constructed to link the wavelet coefficients of the same orientation and locations at adjacent scales. Moreover, the correlations of wavelet coefficients from a coarser scale to a finer scale are different from that from a finer scale to a coarser scale. There are two types of inter-scale edges—coarse-to-fine and fine-to-coarse. The coarse-to-fine inter-scale correlation is derived based on the statistical result by Simoncelli [19], who analyzed the correlation between the dyadic wavelet coefficients in a coarse scale to those at the same location and orientation at the immediate finer scale in a natural image. The coarse-to-fine inter-scale correlation of wavelet coefficient w_{pi} at a coarse scale and wavelet

coefficient w_c at the immediate finer scale can be represented in terms of minus log-probability as

$$\frac{k_1 w_i^2}{w_{pi}^2}, \quad (3)$$

where k_1 is a parameter. Thus, given the wavelet coefficients w_{pi} at the coarse scale, Eq. (3) gives the minus log probability of the wavelet coefficient w_i at the same location and orientation at the immediate fine scale.

On the other hand, the fine-to-coarse inter-scale correlation is derived from the theoretical result of wavelet singularity, analyzed by Mallat and Hwang [20]. Let w_i and w_{ci} be wavelet coefficients corresponding to the same singularity at different scales. Then, w_i and w_{ci} have the same sign and the correlation, in terms of the ratio of the modulus of wavelet coefficients, from w_{ci} at a fine scale to w_i at the immediate coarser scale can be expressed as

$$\frac{|w_i|}{|w_{ci}|} = \frac{w_i}{w_{ci}} = 2^{\alpha+\frac{1}{2}}, \quad (4)$$

where α is the Lipschitz of the singularity. If the singularity is a step edge, then α is 0. The exponent $\alpha + \frac{1}{2}$ in Eq. (4) depends on how a wavelet is normalized. Here, the wavelet is normalized to have unit 2-norm.¹ Eq. (4) can also be expressed in terms of minus log-probability as

$$k_2 \left(w_i - 2^{\alpha+\frac{1}{2}} w_{ci} \right)^2, \quad (5)$$

where k_2 is a parameter. If the type α of the singularity is known, given the wavelet coefficient at the finer scale, w_{ci} , Eq. (5) gives the minus log-probability of the wavelet coefficient w_i at the coarse scale. Since step edges are the most salient features to be recovered from an image, in this chapter, we set α to 0.

2.2 Intra-scale edges

A coherent or similar structure can be used to leverage the quality of the restoration [2, 21]. This is the principle behind the success of BM3D and the example-based approach in image processing [22]. Many similarity metrics have been proposed to derive the coherent structure, such as the mutual information, the kernel functions, and the Pearson's correlation coefficient. In this chapter, the Pearson's correlation coefficient is modified for some technical concern to derive the intra-scale correlation of random variables X and Y :

$$d(X, Y) = \max \left\{ 0, \frac{\mathbf{E}\{(X - \mu_X)(Y - \mu_Y)\}}{\sigma_X \sigma_Y} \right\} + q, \quad (6)$$

where (μ_X, σ_X) and (μ_Y, σ_Y) are the mean and the standard deviation of X and Y , respectively, and $q > 0$ is the offset, introduced to avoid $d(X, Y) = 0$ in inequality constraints in Eq. (8). The value of $d(X, Y)$ lies in $[q, 1 + q]$, measuring the similarity of

¹ If the wavelet is normalized to have unit 1-norm, then the exponent of Eq. (4) should be α .

Figure 1.
 An illustrative example of how the metric $d_{h,l}$ is measured. There are 12 blocks in a sub-band. All the intra-scale edges between the h -th and the l -th nodes are collected. The coefficients on the h -nodes form the random variable X and those on the l -th nodes form the random variable Y . $d_{h,l}$ is then measured based on Eq. (6).

X and Y . The smaller $d(X, Y)$ is, the more the independence of X and Y and the less likely X and Y have coherence structure. As shown in **Figure 1**, the coherence structure in an image is measured on all the coefficients of intra-scale edges in which endpoints take the same locations within a block in a sub-band.

The intra-scale edges are determined based on the sparsity constraint aiming to preserve the edges in which end nodes have similar values. The number of the edges is determined by the parameter:

$$d(y_h, y_l) |x_h - x_l| \leq r, \tag{7}$$

where $d(y_h, y_l)$ is defined in Eq. (6) and obtained from the observation image, and x_h and x_l are the coefficients at the h -th and l -th nodes, respectively. If the observed values y_h and y_l are similar, the value of $d(y_h, y_l)$ is large, then $|x_h - x_l|$ would be small to satisfy the constraint. This preserves the intensities between x_h and x_h . Only the edges satisfying Eq. (7) are retained in the optimal graph. In the following, $d_{h,l}$ is used to simplify the notion $d(y_h, y_l)$ in Eq. (7).

2.3 Graph construction

The aforementioned are integrated and summarized for our context-dependent representation of an image. An image is divided into blocks. Each block is classified as either a singular block or a smooth block. A singular block is then represented with the dyadic wavelet transform, where the scale is sampled following a geometrical sequence of the ratio of 2 and the spatial domain is not down-sampled. The dyadic wavelet transform of an image is comprised of four sub-bands, LL, LH, HL, and HH, with the last three being the orientation sub-bands. A singular block is associated with

Figure 2.

A block of four pixels can be a smooth block (top) or a singular block (bottom). A smooth block is processed in the image domain. A singular block is in the wavelet domain, where a multi-scale graph is associated with each orientation. The blue and green are built-in directed inter-scale edges and the red are un-directed intra-scale edges. The inter-scale edge connects nodes at the same locations and orientation but at scales next to each other. The green edge is coarse-to-fine, linking a node to its parent, while the blue edge is fine-to-coarse, linking a node to its child. The intra-scale edges are determined by graphical ADMM, which decomposes the node update and edge update in the optimization.

three graphs—one for each orientation sub-bands. Since smooth blocks can be well restored in the image domain, the wavelet transform is not applied to the blocks and a smooth block is associated with a graph in the image domain. Each graph, associated with a singular block or a smooth block, is constructed independently of other graphs. **Figure 2** illustrates an example of graph representation for a block of four pixels.

3. Optimal graphs for noise reduction

The noise reduction problem corresponds to problem (1), where H is the identity matrix. Since each graph is solved independently of the other graphs, the following discussion is focused on one graph. A graph can be associated with a singular block or a smooth block.

3.1 Singular blocks

Let y_i and x_i be the wavelet coefficients associated to the i -th node in a sub-band of the observation image and the original image, respectively. Its parent node is denoted

as $p(i)$ and child node is $c(i)$. Let $z_{h,l}$ be the variable defined on the intra-scale edge that connects the h -th and l -th nodes, which are at the same scale and orientation. The optimal wavelet graph can be derived by solving

$$\begin{cases} \min_{x_i, z_{h,l}} \left[\frac{(y_i - x_i)^2}{2\sigma^2} + \frac{k_1 x_i^2}{x_{p(i)}^2} + k_2 (x_i - \sqrt{2} x_{c(i)})^2 \right] \\ d_{h,l} |z_{h,l}| \leq r, \\ z_{h,l} = x_h - x_l, \end{cases} \quad (8)$$

where r, σ, k_1 , and k_2 are non-negative parameters and the constraints are designed as explained under Eq. (7), where r controls the number of edges in the optimal graph. If a node is at the coarsest scale 2^J , where J is the number of decomposition of the wavelet transform, it does not have a parent node and the second term in the object of Eq. (8) is zero; the third term is set to zero for a node at the finest scale 2^1 as it does not have a child node.

Problem (8) has a convenient matrix representation. Let $x = [x_i]$ and $z = [z_{h,l}]$ be the vectors of variables on nodes and intra-scale edges, respectively. Then, the linear constraints between $z_{h,l}$ and x_h and x_l in Eq. (8) can be expressed as

$$A [x^T z^T]^T = 0, \quad (9)$$

where A is a matrix with elements either $-1, 1$, or 0 . Let $A = [A_x A_z]$. It follows that

$$A [x^T z^T]^T = A_x x + A_z z. \quad (10)$$

Each row of A_x has one element which value is 1 and another element which value is -1 and the rest of value 0 . Meanwhile, each row of A_z has one element of values and the rest of value 0 . Let $\lambda = [\lambda_{h,l}] \geq 0^2$ and μ be the vectors of Lagrangian variables associated to inequality constraints and the equality constraints in Eq. (8), respectively. The augmented Lagrangian of Eq. (8) is

$$\begin{aligned} L_\rho(x, z, \lambda, \mu) = & \sum_i \left[\frac{(y_i - x_i)^2}{2\sigma^2} + \frac{k_1 x_i^2}{x_{p(i)}^2} + k_2 (x_i - \sqrt{2} x_{c(i)})^2 \right] \\ & + \mu^T (A_x x + A_z z) + \frac{\rho}{2} \|A_x x + A_z z\|^2 + \sum_{h,l} \lambda_{h,l} d_{h,l} |z_{h,l}| - r \mathbf{1}^T \lambda, \end{aligned} \quad (11)$$

where $\mathbf{1}$ is a vector with all members being 1 ; and $\rho > 0$ are fixed parameters.

The ADMM algorithm intends to blend the decomposability of dual ascent with the superior convergence properties of the method of multipliers. Here, ADMM is used to derive the optimal graph by separating the node and edge update. Graphical ADMM derives the saddle points of

$$\max_{\lambda \geq 0, \mu} \min_{x, z} L_\rho(x, z, \lambda, \mu) \quad (12)$$

² Let $\lambda = [\lambda_{h,l}]$. Then, $\lambda \geq 0$ if and only if $\lambda_{h,l} \geq 0$ for all h and l .

by iterating the following updates:

$$\begin{cases} x^{k+1} = \arg \min_x L_\rho(x, z^k, \lambda^k, \mu^k); \\ z^{k+1} = \mathbb{Q} \left(\arg \min_z L_\rho(x^{k+1}, z, \lambda^k, \mu^k) \right); \\ \lambda_{h,l}^{k+1} = \mathbb{P}_{\geq 0} \left(\lambda_{h,l}^k - \varepsilon \left[r - d_{h,l} |z_{h,l}^{k+1}| \right] \right); \\ \mu^{k+1} = \mu^k + \rho (A_x x^{k+1} + A_z z^{k+1}); \end{cases} \quad (13)$$

where $\mathbb{Q}$ and $\mathbb{P}_{\geq 0}$ are the orthogonal projections to satisfy constraints $d_{j,l} \|z_{h,l}\|_1 \leq r$ and $\lambda_{h,l} \geq 0$, respectively; and $\varepsilon > 0$ is the stepping size for the dual ascent of Lagrangian variables λ . The first and second updates in Eq. (13) update the node variables and edge variables, respectively. The update of the dual variables λ is derived based on the necessary conditions at an optimum of Eq. (8) that

$$\begin{cases} \lambda_{h,l} \geq 0; \\ \lambda_{h,l} (r - d_{h,l} |z_{h,l}^{k+1}|) = 0. \end{cases} \quad (14)$$

The third update in Eq. (13) has the following interpretation. If $(r - d_{h,l} |z_{h,l}^{k+1}|) > 0$, then $\lambda_{h,l}^k$ will decrease and keep its value to be non-negative by $\mathbb{P}$. The value of $\lambda_{h,l}$ can be repeatedly decreased by increasing the iteration number k until either $\lambda_{h,l} = 0$ or $(r - d_{h,l} |z_{h,l}^{k+1}|) = 0$, where the edge is active, and the optimal conditions (14) satisfy. At optimum, either the Lagrangian associated with an edge is zero or the constraint on the edge is active. Only the active edges are retained in the graph. Since the number of active edges is sparse, the edges in the optimal graph are sparse. The solutions for the updates rules for primal variables are derived in Sections 3.1 and 3.2, respectively.

3.2 Smooth blocks

A smooth block is processed in the image domain, where each pixel is associated with a node in a graph. Problem (8) becomes finding the optimal graph by solving

$$\begin{cases} \min_{x_i, z_{h,l}} \sum_i \frac{(y_i - x_i)^2}{2\sigma^2} \\ d_{h,l} |z_{h,l}| \leq r, \\ z_{h,l} = x_h - x_l. \end{cases} \quad (15)$$

The optimal graph can be derived by a method similar to that for a singular block.

3.3 Update edges

The update rule for edge variables z is to solve, with fixed x, λ , and μ ,

$$\begin{cases} \min_z L_\rho(x, z, \lambda, \mu) \\ d_{h,l} |z_{h,l}| \leq r. \end{cases} \quad (16)$$

If only the terms in Eq. (3) relevant to the optimization variables, $\{z_{h,l}\}$, are concerned, Eq. (16) becomes

$$\begin{cases} \min \sum_{h,l} \left(-\mu_{h,l}^T z_{h,l} + \frac{\rho}{2} (x_h - x_l - z_{h,l})^2 + \lambda_{h,l} d_{h,l} |z_{h,l}| \right) \\ |z_{h,l}| \leq \frac{r}{d_{h,l}}, \end{cases} \quad (17)$$

because $d_{h,l}$ is non-zero as defined in Eq. (6). Equation (17) indicates that each edge variable can be updated independently of the others.

We first solve Eq. (17) without the constraint and obtain the solution $u_{h,l}$ of

$$\arg \min_{z_{h,l}} \left(-\mu_{h,l}^T z_{h,l} + \frac{\rho}{2} (x_h - x_l - z_{h,l})^2 + \lambda_{h,l} d_{h,l} |z_{h,l}| \right), \quad (18)$$

by soft-thresholding with

$$u_{h,l} = \begin{cases} x_h - x_l + \frac{\mu_{h,l}}{\rho} - \frac{\lambda_{h,l}}{\rho} d_{h,l}, & \text{if } x_h - x_l + \frac{\mu_{h,l}}{\rho} \geq \frac{\lambda_{h,l}}{\rho} d_{h,l}; \\ x_h - x_l + \frac{\mu_{h,l}}{\rho} + \frac{\lambda_{h,l}}{\rho} d_{h,l}, & \text{if } x_h - x_l + \frac{\mu_{h,l}}{\rho} \leq -\frac{\lambda_{h,l}}{\rho} d_{h,l}; \\ 0, & \text{otherwise.} \end{cases} \quad (19)$$

It is then followed by orthogonally projecting $u_{h,l}$ to satisfy the constraint by solving

$$\begin{cases} \min_{z_{h,l}} \frac{1}{2} (z_{h,l} - u_{h,l})^2 \\ |z_{h,l}| \leq \frac{r}{d_{h,l}}. \end{cases} \quad (20)$$

Eq. (20) can be solved by a sequence of soft-thresholding operations. The algorithm is sketched as follows. First, we check whether $|u_{h,l}| \leq \frac{r}{d_{h,l}}$. If it is, $u_{h,l}$ is the solution. Otherwise, we begin with a small γ and solve

$$z_{h,l}^+ = \arg \min_{z_{h,l}} \frac{1}{2} (z_{h,l} - u_{h,l})^2 + \gamma |z_{h,l}|. \quad (21)$$

The solution is the soft-thresholding as

$$z_{h,l}^+ = \begin{cases} 0, & \text{if } |u_{h,l}| \leq \gamma; \\ \left(1 - \frac{\gamma}{|u_{h,l}|} \right) u_{h,l}, & \text{otherwise.} \end{cases} \quad (22)$$

If $|z_{h,l}^+| \leq \frac{r}{d_{h,l}}$, $z_{h,l}$ is updated to $z_{h,l}^+$, the algorithm stops. Otherwise, γ is increased and Eq. (22) is solved again. Since increasing γ decreases $|z_{h,l}^+|$, this algorithm always stops and updates the edge variable $z_{h,l}$ to meet the constraint.

The complexity to update edge variables is analyzed. If the number of pixels of a block is n and if the dyadic wavelet transform takes J scales, then the number of edge constraints on a singular block is $O(Jn^2)$ and the number of edge constraints on a

smooth block is $O(n^2)$. Let K_1 be the maximum number of iterations to derive the solution for Eq. (20) for all graphs. The complexity of one edge update is $O(K_1 n^2 (|J_F| + 3|J_W|J))$, where 3 is the number of orientations, $|J_F|$ and $|J_W|$ are numbers of singular blocks and smooth blocks, respectively.

3.4 Update nodes

The node update for a singular block is more complicated than that for a smooth block because a graph for a singular block has a multi-scale structure, where adjacent scales are linked by inter-scale edges.

3.5 Singular blocks

To update the nodes x in a singular block is to solve the augmented Lagrangian function (3) via

$$\arg \min_x L_\rho(x, z, \lambda, \mu), \quad (23)$$

for given z, λ , and μ . This is not a convex problem because the second term in Eq. (3) is non-convex.

Our approach is to decompose the problem based on the scale parameter into a sequence of sub-problems. Each scale is associated with two convex sub-problems: one is a coarse-to-fine sub-problem and the other is a fine-to-coarse sub-problem. The coarse-to-fine sub-problem assumes the parent nodes at scale 2^{s+1} were updated earlier, while the fine-to-coarse sub-problem assumes the child nodes at scale 2^{s+1} were updated earlier. Let k be the current iteration number. The coarse-to-fine sub-problem updates the nodes at scale 2^s by minimizing³

$$\sum_i \left[\frac{(y_i - x_i)^2}{2\sigma^2} + \frac{k_1 x_i^2}{[x_{p(i)}^k]^2} \right] + \sum_{h,l} \left[\mu_{h,l}^{k-1} (z_{h,l}^{k-1} - (x_h - x_l)) + \frac{\rho}{2} (z_{h,l}^{k-1} - (x_h - x_l))^2 \right]. \quad (24)$$

On the other hand, the fine-to-coarse sub-problem updates the nodes at scale by minimizing⁴

$$\sum_i \left[\frac{(y_i - x_i)^2}{2\sigma^2} + k_2 \left(x_i - \sqrt{2} x_{c(i)}^k \right)^2 \right] + \sum_{h,l} \left[\mu_{h,l}^{k-1} (z_{h,l}^{k-1} - (x_h - x_l)) + \frac{\rho}{2} (z_{h,l}^{k-1} - (x_h - x_l))^2 \right]. \quad (25)$$

The node update problem (23) can then be approximated by repeatedly solving the coarse-to-fine iteration followed by the fine-to-coarse iteration. The coarse-to-fine iteration solves a sequence of the coarse-to-fine sub-problems beginning at the

³ The second term at below is zero, when 2^s is the coarsest scale.

⁴ The second term below is zero when nodes are at the finest scale.

coarsest scale. In contrast, the fine-to-coarse iteration solves a sequence of the fine-to-coarse sub-problems beginning at the finest scale.

Problems (24) and (25) can be efficiently solved. The objectives in the sub-problems are strictly convex functions because their Hessian matrices are positive definite (as can be observed from the inverse matrix of Eqs. (26) and (27)) and, thus, the optimal solution of each is unique. The closed-form solutions of the sub-problems can be derived as follows.

For convenience, we omit all the superscript index in Eqs. (24) and (25) and let A_x^s and A_z^s denote the sub-matrices of A_x and A_z , respectively. A_x^s and A_z^s retain only the rows and columns in A_x and A_z corresponding to the nodes and edges at scale 2^s , respectively. We also let x_s , z_s , and μ_s denote the vectors of nodes, edges, and Lagrangian variables at scale 2^s . The closed-form solution of x_s of the coarse-to-fine sub-problem is

$$\begin{cases} \left[\frac{1}{\sigma^2} I + C_{s+1}^T C_{s+1} + \rho (A_x^s)^T A_x^s \right]^{-1} \left(\frac{1}{\sigma^2} y_s - (A_x^s)^T (\mu_s - \rho A_z^s z_s) \right) \text{ for } s \neq J; \\ \left[\frac{1}{\sigma^2} I + \rho (A_x^s)^T A_x^s \right]^{-1} \left(\frac{1}{\sigma^2} y_s - (A_x^s)^T (\mu_s - \rho A_z^s z_s) \right) \text{ for } s = J; \end{cases} \quad (26)$$

where C_{s+1} is a diagonal matrix which diagonal element at (i, i) is $\frac{\sqrt{k_1}}{\sqrt{2} \|x_{p(i)}\|}$ and 2^J is the coarsest scale. On the other hand, the closed-form solution of x_s for the fine-to-coarse sub-problem is

$$\begin{cases} \left[\left(\frac{1}{\sigma^2} + 2k_2 \right) I + \rho (A_x^s)^T A_x^s \right]^{-1} \left(\frac{1}{\sigma^2} y_s + 2\sqrt{2} k_2 x_{s-1} - (A_x^s)^T (\mu_s - \rho A_z^s z_s) \right) \text{ for } s \geq 2; \\ \left[\frac{1}{\sigma^2} I + \rho (A_x^s)^T A_x^s \right]^{-1} \left(\frac{1}{\sigma^2} y_s - (A_x^s)^T (\mu_s - \rho A_z^s z_s) \right) \text{ for } s = 1; \end{cases} \quad (27)$$

The complexity of the matrix inversion in Eqs. (26) and (27) is low since A_x^s is a sparse matrix and each row of A_x^s has at most one 1 and one -1 and the rest are zero. The complexity of the sparse matrix inversion in Matlab is proportional to the number of non-zero elements in the matrix. Thus, one iteration of either coarse-to-fine or fine-to-coarse of a graph takes the complexity $O(Jn)$, where J is the number of decomposition and n is the number of pixels at a scale.

3.6 Smooth blocks

The node update for a smooth block can be analytically derived from the problem (15) at the condition that $\{z_{h,l}\}$ is given. If the superscript index is omitted, the closed-form solution is

$$\left[\frac{1}{\sigma^2} I + \rho A_x^T A_x \right]^{-1} \left(\frac{1}{\sigma^2} y - (A_x^s)^T (\mu - \rho A_z z) \right), \quad (28)$$

where A_x and A_z are defined in Eq. (10). The complexity of the inversion of the sparse matrix $\frac{1}{\sigma^2} I + \rho A_x^T A_x$ is propositional to the non-zero elements in the matrix, which is $O(n)$, where n is number of pixels in a block.

If an image has $|J_W|$ singular blocks and $|J_F|$ smooth blocks, and if J is the number of wavelet decompositions, the total complexity of node updates of the image is $O(n(3K_2J|J_W| + |J_F|))$, where K_2 is the maximum number of iterations of coarse-to-fine and fine-to-coarse node update for a singular block.

Graphical ADMM consists of a sequence of updating the primal and dual variables and the complexity of the algorithm is dominated by the primal variable updates. Our analysis of one iteration of Eq. (13) for node updates and edge updates indicates that the costs are $O(n(3K_2J|J_W| + |J_F|))$ and $O(K_1n^2(3|J_W|J + |J_F|))$, respectively, where n is the number of pixels in a block.

4. Optimal graphs for image restoration

There are various image restoration methods [23]. Here, we use the proximal approach proposed in [24] and [12]. The method smartly reduces the image restoration problem into a sequence of noise reduction problems. Since graphical ADMM for noise reduction is efficient, it can be adopted to derive the optimal graphs for image restoration. Like for noise reduction, a graph is handled independently of the other graphs. The following discussion is focused on deriving the optimal graph for a block.

Let $h(x)$ be the objective function

$$\frac{1}{2} \|y - Hx\|^2, \quad (29)$$

with a known blur kernel H ; x_0 is the vector of the current restored image. The proximity function is defined as

$$d_H(x, x_0) = \frac{\beta}{2} \|x - x_0\|^2 - \frac{1}{2} \|Hx - Hx_0\|^2. \quad (30)$$

The parameter β is chosen so that $d_H(x, x_0)$ is strictly convex with respect to x . This implies that its Hessian $\beta I - H^T H$ is a positive definite matrix, which can be achieved by choosing $\beta > \lambda_{\max}(H^T H)$ (the maximal eigenvalue of the matrix $H^T H$). The proximal objective is defined as

$$\tilde{h}(x, x_0) = h(x) + d_H(x, x_0). \quad (31)$$

Simplifying the above objective, we have the following simpler form by removing $\|Hx\|$ from the proximal objective as

$$\tilde{h}(x, x_0) = \frac{\beta}{2} \left\| x - \left(x_0 + \frac{1}{q} H^T (y - Hx_0) \right) \right\|^2 + K \quad (32)$$

where K contains terms unrelated to x . Since x_0, q, H, y are given, the proximal objective can be regarded as a noise reduction problem with the observation vector, $x_0 + \frac{1}{q} H^T (y - Hx_0)$. Thus, $\tilde{h}(x, x_0)$ can be the first term in noise reduction problem (8) and the algorithm for noise reduction can be used to derive the optimal

graph via separating the node and edge updates following Eq. (13) and procedures in Section 3.

5. Experiments and comparisons

We consider several image denoising and deblurring scenarios used as the benchmarks in state-of-the-art algorithms for performance evaluations and comparisons. The setting of experiments is given as follows. The experiments were conducted on images in Sets I and II in **Figure 3**. Set I contains six gray-scaled natural images, Einstein, Boat, Barbara, Lena, Cameraman, and House. The size of each image is 512×512 or 256×256 , downloaded from the USC-SIPI image database [25]; and Set II contains six gray-scaled textures. Some of them were taken from the Brodatz texture set. Through all experiments, each image is divided into 16 equal-sized blocks. A singular block is decomposed into four scales dyadic wavelet transform with the CDF 9/7 wavelet filters. Since the CDF 9/7 filters are close to orthogonal wavelet filters, the noise variance at any sub-band can be set to σ^2 , the variance of noise in the image domain [7].

5.1 Noise reduction performance

Our noise reduction performance was compared against that of BM3D, WBN, and DRUNet. The perceptual quality of the methods is shown in **Figure 4**. The Lena image of BM3D over-smooths the highlighted area of hat, which is rich in edges and textures. Similarly, textures in the highlighted area of hat in DRUNet are smooth. The image of WBN, on the contrary, under-smooths the highlighted smooth area around the chin and shoulder of Lena. These artifacts have been amended by graphical ADMM, as shown in **Figure 4f**.

The quantity comparisons, measured by the peak-signal-to-noise ratio (PSNR), of Set I and Set II are shown in **Tables 1** and **2**, respectively. The testing environments were images contaminated with the noise of variances, σ^2 . As shown, the deep learning-based method (DRUNet) achieves the highest score in almost all environments. However, in the optimization-based methods (BM3D, WBN, proposed), graphical ADMM achieves unanimously the highest score in all environments.

Figure 3.
The images in set I (first row) and set II (second row). Set I contains six gray-scaled natural images and set II contains six gray-scaled textures.

Figure 4. Comparisons of the denoised Lena images derived by BM3D, WBN, DRUNet, and graphical ADMM. The noise standard deviation is set at $\sigma = 25$; (a) the original 512×512 Lena image; (b) the noisy image; (c) the result of BM3D; (d) the result of WBN; (e) the result of DRUNet; and (f) the result of graphical ADMM. Graphical ADMM preserves both the smooth and edged areas in the original image, as shown in the highlighted areas.

5.2 Image restoration performance

Table 3 presents five-point spread functions (PSFs) used for image restoration in literature [11]. Each PSF was normalized to have unit 1-norm before it was used to

Image	Method	PSNR					
		$\sigma = 10$	$\sigma = 15$	$\sigma = 20$	$\sigma = 25$	$\sigma = 30$	$\sigma = 35$
Einstein	BM3D	34.4392	33.0331	32.1694	31.4186	30.8709	30.3777
	WBN	34.4848	33.0821	32.3429	31.4728	30.9178	30.4109
512 × 512	Proposed	34.5013	33.1005	32.3544	31.4862	30.9306	30.4255
	DRUNet	34.9948	33.6019	32.7411	32.1092	31.5952	31.1622
	BM3D	33.8883	32.1067	30.8554	29.8356	29.0954	28.2992
Boat	WBN	33.9095	32.1369	30.8874	29.8561	29.1273	28.3285
512 × 512	Proposed	33.9241	32.1504	30.9063	29.8772	29.1416	28.3410
	DRUNet	34.4264	32.7123	31.5194	30.5768	29.8391	29.1826
	BM3D	34.9567	33.0666	31.7376	30.7176	29.7049	28.8879
Barbara	WBN	34.9643	33.0831	31.7515	30.7332	29.7233	28.4571
512 × 512	Proposed	34.9704	33.1012	31.7735	30.7468	29.7458	28.4675
	DRUNet	35.2115	33.4389	32.1951	31.2341	30.4275	29.7520
	BM3D	36.6367	34.8782	33.0567	32.5501	31.6531	31.0301
Lena	WBN	36.6354	34.8886	33.3048	32.4488	31.5617	30.9148
512 × 512	Proposed	36.6447	34.8960	33.3065	32.5632	31.6744	31.1022
	DRUNet	36.4431	34.9269	33.8363	32.9669	32.2285	31.6072
	BM3D	34.1355	31.8449	30.3797	29.4118	28.5516	27.8758
Cameraman	WBN	34.1637	31.8675	30.5629	29.5723	28.7274	28.0487
256 × 256	Proposed	34.1668	31.8776	30.5732	29.5804	28.7335	28.0614
	DRUNet	34.9927	32.9133	31.5788	30.6079	29.8462	29.2131
	BM3D	36.6638	34.9028	33.7349	32.9084	32.1240	31.5103
House	WBN	36.8538	34.9302	33.7620	32.9363	32.1571	31.5390
256 × 256	Proposed	36.8665	34.9367	33.7742	32.9420	32.1643	31.5404
	DRUNet	37.4420	35.8267	34.7084	33.9251	33.2517	32.6733

Table 1.
Comparisons of the PSNRs of the noise reduction methods on noisy images with the noise of standard deviation σ in set I.

Image	Method	PSNR					
		$\sigma = 10$	$\sigma = 15$	$\sigma = 20$	$\sigma = 25$	$\sigma = 30$	$\sigma = 35$
F0	BM3D	32.5200	30.3213	28.8595	27.7279	26.7751	25.9243
	WBN	32.6212	30.4143	28.9314	27.8087	26.8644	26.0136
512 × 512	Proposed	32.6338	30.4365	28.9572	27.8276	26.8867	26.0345
	DRUNet	33.7186	31.7164	30.2770	29.1838	28.3420	27.6675
	BM3D	29.3575	26.3832	24.3540	22.8154	21.6195	20.6102
F3	WBN	29.5361	26.5738	24.5363	23.0236	21.8514	20.9144

Image	Method	PSNR					
		$\sigma = 10$	$\sigma = 15$	$\sigma = 20$	$\sigma = 25$	$\sigma = 30$	$\sigma = 35$
512×512	Proposed	29.5464	26.5896	24.5554	23.0467	21.8722	20.9364
	DRUNet	30.0832	27.2371	25.3141	23.8812	22.7516	21.8235
	BM3D	29.8987	27.3043	25.6216	24.3522	23.4010	22.5878
F7	WBN	30.1253	27.3765	25.6782	24.4186	23.4862	22.6574
512×512	Proposed	30.1470	27.3932	25.6914	24.4375	23.5002	22.6786
	DRUNet	30.8467	28.2704	26.5578	25.3201	24.3751	23.6274
	BM3D	34.0075	32.4024	31.2442	30.311	29.5213	28.7919
g3	WBN	34.0384	32.4566	31.2517	30.3472	29.5506	28.8064
512×512	Proposed	34.0415	32.4734	31.2570	30.3613	29.5712	28.8115
	DRUNet	34.2632	32.7016	31.6367	30.7665	30.0146	29.3556
	BM3D	31.3155	29.0087	27.4678	26.3278	25.4362	24.7034
p3	WBN	31.3385	29.0274	27.4936	26.3414	25.4535	24.7153
512×512	Proposed	31.3520	29.0475	27.5102	26.3588	25.4763	24.7274
	DRUNet	31.9992	29.7159	28.1847	27.0464	26.1412	25.3947
	BM3D	31.5652	29.5789	28.3481	27.4696	26.7822	26.2039
r3	WBN	31.6362	29.6675	28.4277	27.5486	26.8774	26.2745
512×512	Proposed	31.6521	29.6836	28.4672	27.5665	26.9013	26.3020
	DRUNet	31.8651	29.9371	28.7082	27.8162	27.1131	26.5323

Table 2.
Comparisons of the PSNRs of the noise reduction methods on set II texture images.

Blur Kernel	Formulation	size
h_1	$\frac{1}{1+d_x^2+d_y^2}$	15×15
h_2	Uniform	9×9
h_3	$\begin{bmatrix} 1 \\ 4 \\ 6 \\ 4 \\ 1 \end{bmatrix} \times [1, 4, 6, 4, 1]$	5×5
h_4	Gaussian ($\sigma = 1.6$)	25×25
h_5	Gaussian ($\sigma = 0.4$)	25×25

Table 3.
Blur kernels for experiments. The d_x and d_y in h_1 are, respectively, the horizontal and vertical distances of a pixel to the center of the blur kernel.

blur an image. The performance was compared with the state-of-the-art methods, IDD-BM3D and DPIR. To have fair comparisons, both methods used the same initial images in each experiment. The visual quality of the restored images is shown in **Figure 5** and the blue and red boxes are magnifications of the highlighted areas in the

Figure 5. Comparisons of the deblurred images. The blue and red boxes are the magnified areas in the image. (a) the original 512×512 boat image; (b) the blurred image with blur kernel h_4 ; (c) the image of IDD-BM3D; (d) the image of DPIR; and (e) the image of graphical ADMM. The overall perceptual quality of our image is better since that of IDD-BM3D and DPIR are over-smoothed.

image. Compared with the original images, the overall perceptual quality of the images of IDD-BM3D and DPIR appear over-smoothed, whereas graphical ADMM preserves more image details, leading to better perceptual quality. Graphical ADMM can preserve more details because it uses the multi-scale approach in treating the texture and edge regions. The wavelet persistence property allows information at coarse scales to pass to fine scales and vice versa. As a result, graphical ADMM yields shaper results in recovering singular points in images.

The quantity comparison is shown in **Figure 6**, where the performance improvement of graphical ADMM over IDD-BM3D was measured by the ISNR (increased signal-to-noise ratio) [26]. The ISNR quantitatively assesses the restored images with known ground truths. Let y , x , and x_0 be the vector representations of the observation, the restored image, and the ground truth, respectively; the ISNR is defined as

Figure 6.

Average and standard deviation of the ISNR gain of DPIP over IDD-BM₃D and graphical ADMM over IDD-BM₃D. Each image is blurred and then added to white noise of standard deviation indicated by the noise level. The image was then deblurred. An ISNR gain was calculated from the de-blurred images. The circled point and bar of a measurement at a noise level are the average and standard deviation, respectively, of thirty ISNR gains of natural images from set I ((a) DPIP over IDD-BM₃D and (b) graphical ADMM over IDD-BM₃D) and texture images from set II ((c) DPIP over IDD-BM₃D and (d) graphical ADMM over IDD-BM₃D). As shown, the curves of ISNR gain increase steadily and progressively when the noise level increases.

$$10 \log_{10} \frac{\|y - x_0\|^2}{\|x - x_0\|^2}. \quad (33)$$

The higher the ISNR value of a restored image, the better the restoration quality of the image. The ISNR gain of graphical ADMM over that of IDD-BM₃D is defined as

$$\text{ISNR}(\text{graphical ADMM}) - \text{ISNR}(\text{IDD-BM}_3\text{D}), \quad (34)$$

and the ISNR gain of DPIP over that of IDD-BM₃D is defined as

$$\text{ISNR}(\text{DPIP}) - \text{ISNR}(\text{IDD-BM}_3\text{D}). \quad (35)$$

Figure 6a and **c** show the ISNR gain of DPIP over IDD-BM₃D in Set I and Set II, respectively. **Figure 6b** and **d** show the ISNR gain of graphical ADMM over IDD-BM₃D in Set I and Set II, respectively. Let us take **Figure 6b** as an example. At a noise level, each image in Set I was first blurred by a kernel in **Table 3**. The result was added to white noise to obtain a noisy blurred image. This procedure generated thirty noisy blurred images since Set I contains six images and **Table 3** has five blur kernels. Each noisy blurred image was deblurred. The ISNR gain of the image obtained by graphical ADMM and that by IDD-BM₃D was calculated. The thirty ISNR gains were then used to calculate the mean and standard derivation, as shown in **Figure 6**. The mean ISNR

gain of graphical ADMM increases steadily and progressively over IDD-BM3D, as the noise level increases.

5.3 Discussions

DRUNet and DPIR are deep learning methods and the training data with a noise level of σ ranges from 0 to 50. In the experiments, they have the best performance in quantity comparisons but graphical ADMM is the best in visual quality. For learning-based methods, the training data is important and related to the performance. The inferred results are data-driven and not interpretable. If the training data is less or the distribution of the testing data is not similar to training, the performance will be worse. If the artifacts occurred in the results, we do not know how it happened because the network is just composed of many coefficients trained from the training data. At the same time, the time cost for training is very high. These are all the drawbacks of learning-based methods. However, the optimization-based methods are not limited to the training data. The results are derived from the objective function and interpretable. In addition, they are more stable in practical applications. So, there is a trade-off between learning and optimization-based methods.

For the optimization-based methods, the experiments have demonstrated the advantages of graphical ADMM in both the noise reduction and image restoration tasks over the compared methods. Recall that BM3D and IDD-BM3D adopt the image-dependent tight frame representations. IDD-BM3D also combines the analytic and synthetic optimization methods by de-coupling the noise reduction problem and the image restoration problem. This yields a game-theoretical approach that two formulations are used to minimize a single objective function. The solution adopted by IDD-BM3D is a Nash equilibrium point. The WBN represents an image as a multi-scale probabilistic DAG and adopts the belief propagation to derive the MAP solution.

The advantages of graphical ADMM lie in the context-dependent decompositions of an image horizontally in space and vertically along the scales in handling the image details. The spatial decomposition allows our method to overcome the cons of under-smoothing the smooth areas in WBN and keeps the pros of WBN that preserves sharp edges. Meanwhile, graphical ADMM is much more efficient than the time-consuming belief propagation adopted in WBN.

The mixture of data-dependent and data-independent edges in wavelet graph construction is a significant feature of our method. The intra-scale edges are determined by a data-dependent adaptive process, which imposes sparseness by keeping the edges which end nodes have similar coefficients in the optimal graph. The inter-scale edges are data-independent, built-in to leverage the wavelet persistence property. The inter-scale edges, passing information of singularities from finer scales to coarser scales and vice versa, can preserve more texture and edges in original images. This distinguishes our algorithm from BM3D and IDD-BM3D, which encode structure in atoms of a dictionary and select a few atoms for image representation.

6. Conclusions

We present a novel approach by combining spatial decomposition, vertical (multi-scale) decomposition, and ADMM optimization in a graphical framework for image

noise reduction and restoration tasks. The graphical ADMM method has demonstrated that its results are superior to those of state-of-the-art algorithms. We also demonstrated that mixing data-dependent and data-independent structures in a graph representation can leverage the sparseness and persistence of a wavelet representation. Rather than adopting a combinatorial approach to derive an optimal graph, we showed that the graph can be derived by a numerically tractable optimization approach. In addition, we showed that the optimization problem is well coupled with our graph representation, and can be decomposed into a sequence of convex sub-problems, with each having an efficient closed-form solution. This opens a new perspective of combining a mixture of data-adaptive and data-independent structures, hierarchical decomposition, and optimization algorithms in modeling, representing, and solving more image processing tasks.

Author details

Jinn Ho, Shih-Shuo Tung and Wen-Liang Hwang*
Institute of Information Science, Academia Sinica, Taiwan

*Address all correspondence to: whwang@iis.sinica.edu.tw

IntechOpen

© 2022 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

References

- [1] Rudin L, Osher S, Fatemi E. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*. 1992;**60**(1–4):259-268. DOI: 10.1016/0167-2789(92)90242-F
- [2] Buades A, Coll B, Morel J. A review of image denoising algorithms, with a new one. *Multiscale Modeling & Simulation*. 2005;**4**(2):490-530. DOI: 10.1137/040616024
- [3] Elad M, Aharon M. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image Processing*. 2006;**15**(12):3736-3745. DOI: 10.1109/TIP.2006.881969
- [4] Chatterjee P, Milanfar P. Is denoising dead? *IEEE Transactions on Image Processing*. 2010;**19**(4):895-911. DOI: 10.1109/TIP.2009.2037087
- [5] Li J, Shen Z, Yin R, Zhang X. A reweighted l2 method for image restoration with poisson and mixed poisson-gaussian noise. *Inverse Problem and Imaging*. 2015;**9**(3):875-894. DOI: 10.3934/ipi.2015.9.875
- [6] Dabov K, Foi A, Katkovnik V, Egiazarian K. Image denoising by sparse 3-D transform-domain collaborative filtering. *IEEE Transactions on Image Processing*. 2007;**16**(8):2080-2095. DOI: 10.1109/TIP.2007.901238
- [7] Ho J, Hwang W. Wavelet Bayesian network image denoising. *IEEE Transactions on Image Processing*. 2013;**22**(4):1277-1290. DOI: 10.1109/TIP.2012.2220150
- [8] Zhang K, Li Y, Zuo W, Zhang L, Gool L, Timofte R. Plug-and-play image restoration with deep denoiser prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2021. DOI: 10.1109/TPAMI.2021.3088914
- [9] Ronneberger O, Fischer P, Brox T. U-net: Convolutional networks for biomedical image segmentation. *International Conference on Medical Image Computing and Computer-Assisted Intervention*. 2015:234-241. DOI: 10.1007/978-3-319-24574-4_28
- [10] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. *IEEE Conference on Computer Vision and Pattern Recognition*. 2016:770-778. DOI: 10.1109/CVPR.2016.90
- [11] Danielyan A, Katkovnik V, Egiazarian K. BM3D frames and variational image deblurring. *IEEE Transactions on Image Processing*. 2012;**21**(4):1715-1728. DOI: 10.1109/TIP.2011.2176954
- [12] Zibulevsky M, Elad M. L1-L2 optimization in signal and image processing. *IEEE Signal Processing Magazine*. 2010;**27**(3):76-88. DOI: 10.1109/MSP.2010.936023
- [13] Heckerman D. A Tutorial on learning bayesian networks. In: *Innovations in Bayesian Networks*. Berlin/Heidelberg: Springer; 2008. pp. 33-82. DOI: 10.1007/978-3-540-85066-3_3
- [14] Chickering D, Heckerman D, Meek C. A Bayesian approach to learning Bayesian networks with local structure. In *Proceedings of Thirteenth Conference on Uncertainty in Artificial Intelligence*. 1997:80-89. Available from: <https://dl.acm.org/doi/pdf/10.5555/2074226.2074236>
- [15] Cai J, Osher S, Shen Z. Split Bregman methods and frame based image restoration. *Multiscale Modeling &*

Simulation. 2009;8(2):337-369.
DOI: 10.1137/090753504

[16] Boyd S, Parikh N, Chu E, Peleato B, Eckstein J. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*. 2011;3(1):1-122. DOI: 10.1561/22000000016

[17] Hoiem D, Efros A, Hebert M. Geometric context from a single image. *IEEE International Conference on Computer Vision*. 2005;1:654-661. DOI: 10.1109/ICCV.2005.107

[18] Ji H, Luo Y, Shen Z. Image recovery via geometrically structured approximation. *Applied and Computational Harmonic Analysis*. 2016;41(1):75-93. DOI: 10.1016/j.acha.2015.08.012

[19] Simoncelli E. Bayesian denoising of visual images in the wavelet domain. *Bayesian Inference in Wavelet Based Models*. 1999;141:291-308. DOI: 10.1007/978-1-4612-0567-8_18

[20] Mallat S, Hwang W. Singularity detection and processing with wavelets. *IEEE Transactions on Information Processing*. 1992;38(2):617-643. DOI: 10.1109/18.119727

[21] Milanfar P. A tour of modern image filtering: New insights and methods, both practical and theoretical. *IEEE Signal Processing Magazine*. 2013;30(1):106-128. DOI: 10.1109/MSP.2011.2179329

[22] Sreedevi P, Hwang W, Lei S. An exemplar-based approach for texture compaction synthesis and retrieval. *IEEE Transactions on Image Processing*. 2010;19(5):1307-1318. DOI: 10.1109/TIP.2009.2039665

[23] Mairal J, Elad M, Sapiro G. Sparse representation for color image restoration. *IEEE Transactions on Image Processing*. 2008;17(1):53-69. DOI: 10.1109/TIP.2007.911828

[24] Daubechies I, Defrise M, De-Mol C. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Communications on Pure and Applied Mathematics*. 2004;57(11):1413-1457. DOI: 10.1002/cpa.20042

[25] The USC-SIPI Image Database, Signal and Image Processing Institute, University of Southern California. Available from: <https://sipi.usc.edu/database/>

[26] Hanif M, Seghouane A. Blind image deblurring using non-negative sparse approximation. *IEEE International Conference on Image Processing*. 2014. DOI: 10.1109/ICIP.2014.7025821