Image Recovery by Decomposition with Component-Wise Regularization

Shunsuke ONO†, Student Member, Takamichi MIYATA†, Member, Isao YAMADA†, and Katsunori YAMAOKA†, Senior Members

SUMMARY Solving image recovery problems requires the use of some efficient regularizations based on a priori information with respect to the unknown original image. Naturally, we can assume that an image is modeled as the sum of smooth, edge, and texture components. To obtain a high quality recovered image, appropriate regularizations for each individual component are required. In this paper, we propose a novel image recovery technique which performs decomposition and recovery simultaneously. We formulate image recovery as a non-smooth convex optimization problem and design an iterative scheme based on the alternating direction method of multipliers (ADMM) for approximating its global minimizer efficiently. Experimental results reveal that the proposed image recovery technique outperforms a state-of-the-art method.

key words: image recovery, decomposition, regularization, sparsity, convex optimization, alternating direction method of multipliers (ADMM)

1. Introduction

Many image recovery problems can be posed as the inversion of the linear system

\[ \mathbf{v} = \mathbf{F} \mathbf{u}_{\text{org}} + \eta, \] (1)

where \( \mathbf{u}_{\text{org}} \in \mathbb{R}^N \) is the unknown original image with \( N = n_x \times n_y \) pixels, \( \mathbf{v} \in \mathbb{R}^{M \leq N} \) is a known degraded observation, \( \eta \in \mathbb{R}^M \) is an unknown additive noise term (e.g., white Gaussian noise), and \( \mathbf{F} : \mathbb{R}^N \rightarrow \mathbb{R}^M \) is a known linear degradation operator.

Because of the existence of small singular values of \( \mathbf{F} \) and/or the existence of an additive noise, this inverse problem cannot be solved satisfactorily by a standard least-squares approach (for example by the pseudo-inverse of \( \mathbf{F} \)). To obtain a better approximation of \( \mathbf{u}_{\text{org}} \), the so-called regularization is employed in many optimization scenarios by introducing penalty terms. The regularization utilizes a prior knowledge relevant to the unknown original signal. In the last decade, regularization approaches focusing on the sparsity of signals have gained considerable attention within convex optimization frameworks [12]–[14], [23].

Particularly in the field of image recovery, regularization approaches based on the total variation (TV) semi-norm and \( \ell_1 \) norm of certain frame coefficients have been widely considered as powerful choices. These approaches restore an image by solving convex optimization problems, and most of them can be dealt with the following problem formulation.

**Problem 1.1:** Let \( \mathbf{u}_1, \ldots, \mathbf{u}_m \in \mathbb{R}^N \) be components which satisfy \( \sum_{k=1}^m \mathbf{u}_k = \mathbf{u} \) (\( \mathbf{u} \) denotes a recovered image), and \( F_1, \ldots, F_p \) be proper lower semicontinuous convex functions (see Appendix A) from \( \mathbb{R}^N \rightarrow \mathbb{R}^N \) to \((-\infty, \infty] \), each of which the proximity operator (explained in Sect. 2) is available with reasonably low computational cost. Then the optimization problem is as follows:

\[ \min_{(\mathbf{u}_1, \ldots, \mathbf{u}_m) \in \mathbb{R}^N} \sum_{k=1}^m F_k(\mathbf{u}_1, \ldots, \mathbf{u}_m). \] (2)

In the case of \( m = 1 \) (i.e., \( \mathbf{u}_1 = \mathbf{u} \)) and \( p = 2 \), the problem (2) is a general formulation of standard regularization approaches which employ one regularization function and one fidelity control function. This type of formulation is found, for example, in [1], [3], and [26].

In the case of \( m = 1 \) and \( p \geq 3 \), the problem (2) utilizes multiple regularization functions. Many image recovery techniques use this type of formulation [2], [16], [21], [22], [24], [25].

In the case of \( m \geq 2 \) and \( p \geq 3 \), the problem (2) applies multiple regularization functions individually to each component \( \mathbf{u}_1, \ldots, \mathbf{u}_m \). In [7], geometry and texture components are restored from a degraded observation (i.e., \( m = 2 \)). Similar decomposition has been studied in [4], [12], and [28].

Intuitively, it is very natural to assume that an image consists of the sum of the following three conflicting components: smooth, edge, and texture components. These components are characterized by the following properties:

- **Smooth component** consists of almost flat areas with similar values or smooth gradation.
- **Edge component** consists of lines and contours that possess strong directionalities.
- **Texture component** consists of repetitive patterns with local fluctuation.

For brevity, we refer to these three as SET (Smooth, Edge, and Texture) components.

Under the assumption above, in this paper, we propose an image recovery technique called “decomposed SET (D-SET) recovery” with a novel use of multiple regularization functions. First, we introduce an optimization problem in...
the form of (2) associated with multiple regularization functions which take the property of SET components into consideration (i.e., \( m = 3 \)). Then, we reformulate the optimization problem into a standard form suitable for a convex optimization algorithm called the alternating direction method of multipliers (ADMM) [15]. Finally, the proposed ADMM-based iterative scheme which efficiently approximates the solution of the optimization problem is presented. Hence, the proposed method is expected to achieve an effective image recovery with simultaneous reconstruction of conflicting components.

This paper is organized as follows. Section 2 presents basic mathematical tools utilized in our proposed method. Then, we propose D-SET recovery in Sect. 3. In Sect. 4, the efficacy of D-SET recovery is demonstrated through some numerical experiments. Finally, in Sect. 5, we conclude the paper with some remarks.

2. Preliminaries

2.1 Proximity Operator

In this paper, we use the notion of proximity operator which was introduced originally by Moreau in 1962 [20]. For elementary terminologies in convex analysis, see Appendix A.

**Definition 2.1** (Proximity operator): Let \( \Gamma_0(\mathbb{R}^N) \) be the class of proper lower semicontinuous convex functions on \( \mathbb{R}^N \). The proximity operator of index \( \gamma \in (0, \infty) \) of \( h \in \Gamma_0(\mathbb{R}^N) \) is defined by

\[
\text{prox}_h : \mathbb{R}^N \to \mathbb{R}^N : \mathbf{y} \mapsto \arg\min_{\mathbf{x} \in \mathbb{R}^N} \left\{ h(\mathbf{x}) + \frac{1}{2\gamma} ||y-x||_2^2 \right\},
\]

where the existence and uniqueness of the minimizer are guaranteed respectively by the coercivity and strict convexity of \( h(\cdot) + 1/2\gamma||\cdot||_2^2 \).

Here, \( ||\cdot||_2 \) denotes the \( \ell_2 \) Euclidean norm. The proximity operator has been widely employed in image and signal processing [14], [31]. The computational complexity of the proximity operator depends on the function \( h \). We introduce some examples of \( h \) as follows that will be used in our method.

**Example 2.1:** Define \( h : \mathbb{R}^{\gamma N_1 \times N_2} = (\mathbb{R}^N)^{N_1 \times N_2} \to [0, \infty) \) by

\[
h(\mathbf{x}(:= [\mathbf{x}_{ij}]_{ij})) := \sum_{i,j} \sqrt{||\nabla_1 \mathbf{x}_{ij}||_2^2 + ||\nabla_2 \mathbf{x}_{ij}||_2^2},
\]

where \( ||\cdot||_TV \) denotes the total variation (TV) semi-norm, \( \nabla_1, \nabla_2 \in \mathbb{R}^{N \times N} \) are the discrete horizontal and vertical gradient operators defined as follows:

\[
(\nabla_1 \mathbf{x})_{ij} = \begin{cases} x_{i+1,j} - x_{ij}, & \text{if } j < n_h, \\ x_{ij} - x_{i,j}, & \text{if } j = n_h, \end{cases}
\]

(\nabla_2 \mathbf{x})_{ij} = \begin{cases} x_{ij+1} - x_{ij}, & \text{if } j < n_h, \\ x_{ij} - x_{ij}, & \text{if } j = n_h. \end{cases}
\]

In this case, the operation (3) is equal to the computation of the minimization of the well-known TV-based regularization called the ROF model [27]. First, Chambolle [10] proposed an iterative scheme for the computation of the minimizer of the ROF model which is subsequently known as the projected gradient method. Its accelerated version is found in [5] and [33]. Indeed, if (TV(\mathbf{y}, \gamma))_{\gamma \to 0}^\gamma denotes the sequence of the projected gradient method for approximating the minimizer in (3), we have

\[
\text{prox}_{\gamma h}(\mathbf{y}) = \lim_{\gamma \to \infty} \text{TV}(\mathbf{y}, \gamma).
\]

**Example 2.2:** Define \( h : \mathbb{R}^N \to [0, \infty) \) by

\[
h : \mathbf{x} := (x_0, \ldots, x_{N-1})' \mapsto ||\mathbf{x}||_1 := \sum_i |x_i|.
\]

In this case, the proximity operator is simply given by the soft thresholding [18], i.e., for \( i = 0, \ldots, N - 1, \)

\[
[\text{prox}_{\gamma h}(\mathbf{y})]_i = [\text{ST}(\mathbf{y}, \gamma)]_i = \begin{cases} y_i - \gamma, & \text{if } y_i > \gamma, \\ y_i + \gamma, & \text{if } y_i < -\gamma, \\ 0, & \text{if } |y_i| \leq \gamma. \end{cases}
\]

**Example 2.3:** For a given nonempty closed convex set \( C \subset \mathbb{R}^N \), define the indicator function \( \iota_C : \mathbb{R}^N \to [0, \infty) \) by

\[
\iota_C(\mathbf{y}) = \begin{cases} 0, & \text{if } \mathbf{y} \in C, \\ \infty, & \text{otherwise}. \end{cases}
\]

The proximity operator of \( \iota_C \) is given by the metric projection onto \( C \), i.e.,

\[
\text{prox}_{\gamma \iota_C}(\mathbf{y}) = P_C(\mathbf{y}) := \arg\min_{\mathbf{x} \in C} ||y - \mathbf{x}||_2.
\]

2.2 Alternating Direction Method of Multipliers (ADMM)

**Problem 2.1:** Suppose \( \mathbf{G} \in \mathbb{R}^{N_1 \times N_2} \) has full-column rank and let \( f \in \Gamma_0(\mathbb{R}^{N_1}) \) and \( g \in \Gamma_0(\mathbb{R}^{N_2}) \). The problem is to find

\[
(\mathbf{y}^*, \mathbf{z}^*) \in \arg\min_{(\mathbf{y}, \mathbf{z}) \in (\mathbb{R}^{N_1}) \times (\mathbb{R}^{N_2})} f(\mathbf{y}) + g(\mathbf{z}) \quad \text{s.t.} \quad \mathbf{z} = \mathbf{Gy}.
\]

ADMM [15], as shown in Algorithm 2.1, is an iterative scheme which approximates a solution of problem (12). Variants of ADMM are proposed in [32].

**Remark 2.1:** The proof of the objective convergence \( \lim_{k \to \infty} f(\mathbf{y}^{(k)}) + g(\mathbf{z}^{(k)}) = f(\mathbf{y}^*) + g(\mathbf{z}^*) \) and residual convergence \( \lim_{k \to \infty} ||\mathbf{z}^{(k)} - \mathbf{Gy}^{(k)}||_2 = 0 \) of ADMM are found in Sect. 3.2 of [6].

2.3 Discrete Curvelet Transform (DCvT) and Shift-Invariant Redundant Discrete Cosine Transform (RDCT)

The curvelet transform [8] provides an essentially optimal
Algorithm 2.1 (ADMM)
1: Set $k = 0$, choose $z^{(0)}$ and $b^{(0)}$.
2: while a stop criterion is not satisfied do
3: \[ y^{(k+1)} = \arg\min_{y \in \mathbb{R}^n} \left\{ f(y) + \frac{1}{2\eta} || z^{(k)} - Gy - b^{(k)} ||_2^2 \right\} \]
4: \[ z^{(k+1)} = \arg\min_{z \in \mathbb{R}^n} \left\{ g(z) + \frac{1}{2\eta} || z - Gy - b^{(k)} ||_2^2 \right\} \]
5: \[ b^{(k+1)} = b^{(k)} + Gy^{(k+1)} - z^{(k+1)} \]
6: $k \leftarrow k + 1$
7: end while

representation of a certain 2-D function (a continuous image) which is said to be of class $C^2$ except for discontinuities along piecewise $C^2$ curves. In other words, if we denote $f_M$ as the approximation of $f$ which is reconstructed by $M$ largest curvelet coefficients of $f$, the approximation error $\| f - f_M \|_2$ is lower than $cM^{-2} \log_2 M$ ($c$ denotes a certain constant). This error rate is much improved compared to that of the wavelet transform $O(M^{-1})$. For the numerical implementation in this paper, we use the discrete curvelet transform (DCvT) [9] with source-code (CurveLab package [34]) which was proposed as the discrete approximation of the original curvelet transform [8]. If we denote $\Psi \in \mathbb{R}^{N \times N}$ ($N$ is the number of curvelet coefficients) as the DCvT matrix, it satisfies $\Psi^T \Psi = I_N$ where $\Psi$ corresponds to the pseudo inverse DCvT ($I_N \in \mathbb{R}^{N \times N}$ denotes the identity matrix).

DCvT is expected to provide a sparse representation of the edge component because we assumed that the edge component only consists of directional lines and contours, and the other areas in which there is no edge are flat, i.e., the edge component has almost as same property as the function $f$ which was characterized above. For its high level of edge awareness, DCvT is utilized for detecting retinal blood vessels in medical images [19].

The shift-invariant redundant discrete cosine transform (RDCT) is constructed from the orthogonal block discrete cosine transform (DCT). RDCT amounts to applying the orthogonal block DCT to the overlapped sub-blocks of an image $u$, where the size of each block is $B \times B$ and the number of blocks is $n_B \times n_B$. Hence the number of RDCT coefficients becomes $B^2 N$. The RDCT matrix $\Psi_r \in \mathbb{R}^{B^2 \times N}$ can be defined as

\[
\Psi_r = \begin{bmatrix}
    B^{-1} D_{11} \\
    \vdots \\
    B^{-1} D_{n_B \times n_B}
\end{bmatrix},
\]

where $D_{ij} \in \mathbb{R}^{B \times B}$ comprises the bases of the DCT transform matrix corresponding to the $(i,j)$-th block and the other entries are zero. The pseudo inverse RDCT transform is equal to $\Psi_r^T$ which satisfies $\Psi_r^T \Psi_r = I_N$. RDCT is expected to represent the local repetitive patterns of texture more sparsely compared to the so-called scale-shift transforms (e.g. the shift-invariant discrete wavelet transform) which have been used for sparse texture representation (e.g. [24]). One of the reasons is that scale-shift transforms are originally designed for the sparse representation of both edge and texture. On the other hand, if we only consider texture, a block-wise transform like RDCT is expected to easily pick up its associated local features. RDCT is used in the super-resolution decoding of JPEG image in [17].

3. Proposed Method

3.1 Proposed Image Recovery Model

Consider the decomposition of a recovered image $u$ as

\[
u = u_1 + u_2 + u_3,
\]

where $u_1, u_2, u_3$ are the restored components by our method. We call them pseudo SET components. Let $X = \mathbb{R}^N \times \mathbb{R}^N \times \mathbb{R}^N$ be a real Hilbert space where the inner product $\langle \cdot, \cdot \rangle_X : X \times X \to \mathbb{R}$ and its induced norm $\| \cdot \|_X$ are defined as

\[
\langle (x_1, x_2, x_3), (y_1, y_2, y_3) \rangle_X := x_1^T y_1 + x_2^T y_2 + x_3^T y_3,
\]

\[
\| (x_1, x_2, x_3) \|_X := \sqrt{(x_1, x_2, x_3, (x_1, x_2, x_3))}_X,
\]

for $(x_1, x_2, x_3), (y_1, y_2, y_3) \in X$ respectively. In the framework (2), we propose the following nonsmooth convex optimization problem.

Problem 3.1 (D-SET optimization problem):

\[
\min_{(u_1, u_2, u_3) \in X} F(u_1, u_2, u_3) = F_1(u_1) + F_2(\Psi_r u_2) + F_3(\Psi_r u_3)
\]

\[
+ F_4(u_1 + u_2 + u_3) + F_5(\Phi(u_1 + u_2 + u_3)),
\]

where $F_1 : x \mapsto \psi_{TV}(x), F_2 : x \mapsto \psi_{u2}(x), F_3 : x \mapsto \psi_{u3}(x), F_4 : x \mapsto \iota_{\mathbb{R}^B}(x), F_5 : x \mapsto \iota_{\mathbb{R}^B}(x)$.

Remark 3.1 (Roles of the functions in Problem 3.1):

$F_1$: suppressing this regularization function is expected to approximate the smooth component because the TV semi-norm of smooth component becomes small.

$F_2$: suppressing this regularization function is expected to pick up the edge component because DCvT presents the sparse representation of lines and contours.

$F_3$: suppressing this regularization function is expected to decompose the texture component from an image because RDCT presents the sparse representation of the locally repetitive patterns of texture.

$F_4$: this is the constraint on the numerical range of pixels represented by the following nonempty closed convex set

\[
C = \{ x \in \mathbb{R}^B | x_i \in [0, 255] \text{ for } i = 1, \ldots, N \}.
\]

$F_5$: this is the fidelity constraint with respect to $v$ represented by the following nonempty closed convex set

\[
S_{\varepsilon,v} = \{ x \in \mathbb{R}^B | \| x - v \|_2 \leq \varepsilon \},
\]

where $\varepsilon$ is determined by the power of noise.
The functions $F_4$ and $F_5$ constantly take 0 as long as $u \in C$ and $\Phi u \in S_{N, \Psi}$, otherwise they take $\infty$. As a consequence, by minimizing $F$ in (17), the SET components are expected to be restored simultaneously.

**Proposition 3.1:** There exists $(u^*_1, u^*_2, u^*_3) \in X$ which achieves $F(u^*_1, u^*_2, u^*_3) \leq F(u_1, u_2, u_3)$ ($u(u_1, u_2, u_3) \in X$).

The proof of Proposition 3.1 is given in Appendix B.

### 3.2 Reformulation of the D-SET Optimization Problem

In this section, we demonstrate that Problem 3.1 (D-SET optimization problem) can be reformulated into a special case of Problem 2.1. Let $z_1 = u_1 \in \mathbb{R}^N$, $z_2 = \Psi_1 u_2 \in \mathbb{R}^N$, $z_3 = \Psi_2 u_3 \in \mathbb{R}^N$, $z_4 = u_4 + u_5 \in \mathbb{R}^N$, $z_5 = \Phi(u_1 + u_2 + u_3) \in \mathbb{R}^N$, and set $y = [u^*_1, u^*_2, u^*_3, y^*] ! \in \mathbb{R}^M$, $z = [z_1 \ldots z_5] ! \in \mathbb{R}^{2N + N + B^N + M + bN + M + bN}$.

$$G = \begin{bmatrix} I_N & O & O \\ O & \Psi_1 & O \\ I_N & I_N & I_N \\ \Phi & \Phi & \Phi \end{bmatrix} \in \mathbb{R}^{(2N + N + b^N + M + bN + M) \times 3N}. (20)$$

Then, Problem 3.1 can be rewritten in the following form.

**Problem 3.2** (ADMM-applicable form of Problem 3.1):

$$\min_{y \in \mathbb{R}^{3N+2N+b^N+b^N+bN+m}} f(y) + g(z) \quad \text{s.t.} \quad z = Gy, (21)$$

where $f : y \mapsto 0$, $g : z \mapsto \sum_{j=1}^5 F_j(z_j) = w_1||z_1||_TV + w_2||z_2||_1 + w_3||z_3||_1 + \epsilon_c(z_4) + \epsilon_{x_5}(z_5)$.

Problem 3.2 is of the same form as (12) and $G$ is obviously full column rank. Therefore, we can solve the D-SET optimization problem by applying ADMM (Algorithm 2.1) to Problem 3.2.

### 3.3 Iterative Scheme for the D-SET Optimization Problem

Let us explain how to calculate each step of ADMM as applied to Problem 3.2.

First, we consider Step 3. The fact that $f(y) = 0$ turns Step 3 of ADMM into

$$y^{(k+1)} = \arg\min_{y \in \mathbb{R}^N} \frac{1}{2\gamma} ||z^{(k)} - Gy - b^{(k)}||_2^2. (22)$$

Since $G^TG$ is positive-definite by definition of $G$, the minimizer of (22) exists uniquely. This minimizer is obtained by solving a system of linear equations (see Appendix C).

Step 4 of ADMM can be rewritten as

$$\begin{bmatrix} z_1^{(k+1)} \\ \vdots \\ z_5^{(k+1)} \end{bmatrix} = \arg\min_{z_1, \ldots, z_5} \sum_{j=1}^5 \left[ F_j(z_j) + \frac{1}{2\gamma} ||z_j - r_j^{(k+1)}||_2^2 \right]. (23)$$

where $r_1^{(k+1)} = u_1^{(k+1)} + b_1^{(k)}, r_2^{(k+1)} = \Psi_1 u_2^{(k+1)} + b_2^{(k)}, r_3^{(k+1)} = \sum_{j=1}^3 u_j^{(k+1)} + b_3^{(k)},$ and $r_4^{(k+1)} = \sum_{j=1}^3 (\Phi u_3^{(k+1)}) + b_4^{(k)}$ (note $b = [b_1', \ldots, b_5']'$). Evidently, (23) are decoupled with respect to $z_1, \ldots, z_5$. Hence, it can be solved separately as

$$z_j^{(k+1)} = \arg\min_{s_j} \left[ F_j(z_j) + \frac{1}{2\gamma} ||z_j - r_j^{(k+1)}||_2^2 \right] = \text{prox}_{\gamma F_j}(r_j^{(k+1)}) \quad \text{for} \quad j = 1, \ldots, 5. (24)$$

By (9), the proximities operators of $F_2(z_2) = w_2||z_2||_1$ and $F_3(z_3) = w_3||z_3||_1$ can be exactly implemented as

$$z_j^{(k+1)} = \text{prox}_{\gamma F_j}(r_j^{(k+1)}) = ST(r_j^{(k+1)}, w_2y). (26)$$

$$z_3^{(k+1)} = \text{prox}_{\gamma F_3}(r_3^{(k+1)}) = ST(r_3^{(k+1)}, w_3y). (27)$$

Finally, we obtain an ADMM-based iterative scheme for solving the D-SET optimization problem as Algorithm 3.1.

**Remark 3.2** (Implementation of $\text{prox}_{\gamma F_j}$): We approximate the proximity operator of $F_j$ by the accelerated projected gradient method [5] with a finite number of iterations. Our experimental results show that Algorithm 3.1 works well even in the case.

### 4. Numerical Experiments

We demonstrate the efficacy of D-SET recovery by applying it to two image recovery problems and comparing its performance with a state-of-the-art image recovery method.

#### 4.1 Method for Comparison

We compare our method with the decomposed geometry-texture (D-GT) recovery proposed in [7]. The D-GT recovery model is posed as the following optimization problem:
els]) shown in Fig. 1. We employ DCvT with 4 scales and

The ND-SET optimization problem applies the same func-

4.2 Experimental Setting

The two components \( u_g \) and \( u_t \) satisfy \( u_g + u_t = u \). Hence

Additionally, to verify the advantage of SET decompo-

The ND-SET optimization problem applies the same func-

4.2 Experimental Setting

We use eight standard images (256 × 256 = 65,536 [pix-

For complete information about SSIM, see [29]. The iter-

Algorithm 3.1 (Iterative scheme for the D-SET optimization problem)

\[
\begin{align*}
\text{1: } & \text{Set } k = 0 \text{ and choose } w_1, w_2, w_3, \gamma. \\
\text{2: } & \text{for } j = 1 \text{ to } 5 \text{ do} \\
\text{3: } & \quad z_j^{(0)} = 0 \\
\text{4: } & \quad b_j^{(0)} = 0 \\
\text{5: } & \text{end for} \\
\text{6: } & \text{while a stop criterion is not satisfied do} \\
\text{7: } & \quad y^{(k+1)} = \arg\min_{y \in \mathbb{R}^n} \frac{1}{\gamma} \| z^{(k)} - G y - b^{(k)} \|_2^2 \\
\text{8: } & \text{for } j = 1 \text{ to } 5 \text{ do} \\
\text{9: } & \quad z_j^{(k+1)} = \text{prox}_{\gamma G_j} (r_j^{(k+1)}) \text{ (see, (23) - (29))} \\
\text{10: } & \text{end for} \\
\text{11: } & \quad b^{(k+1)} = b^{(k)} + G y^{(k+1)} - z^{(k+1)} \\
\text{12: } & \quad k \leftarrow k + 1 \\
\text{13: } & \text{end while} \\
\min_{(u_g, u_t) \in \mathbb{R}^n \times \mathbb{R}^n} w_g \| u_g \|_V + w_t \| \Psi_t u_t \|_1 + \iota_C (u_g + u_t) \\
+ \iota_E (u_g + u_t) + \iota_{S_c} (\Phi (u_g + u_t)), \quad (30)
\end{align*}
\]

where \( u_g \in \mathbb{R}^n \) and \( u_t \in \mathbb{R}^n \) are the geometry and texture

components, respectively. The matrix \( \Psi_t \in \mathbb{R}^{N \times 2N} \) is the
dual-tree symlet transform matrix with length 6 applied over
3 resolution levels, and \( \iota_C \) is the indicator function of the
nonempty closed convex set \( E_C \) defined as

\[
E_C = \{ x \in \mathbb{R}^N | \| (P x) \|_2 \leq \delta \}, \quad (31)
\]

where \( P \in \mathbb{C}^{N \times N} \) is the discrete Fourier transform matrix.
The two components \( u_g \) and \( u_t \) satisfy

Hence the D-GT optimization problem (30) is a special case of (2)
for \( n = 2 \).

Additionally, to verify the advantage of SET decompo-

The ND-SET optimization problem applies the same functions
employed in the D-SET optimization problem to an image
\( u \) without any decomposition. Thus, the ND-SET optimi-

ization problem can be categorized as an \( m = 1 \) case of
\( (2) \).

The ND-SET optimization problem applies the same func-

\[
\begin{align*}
\min_{u \in \mathbb{R}^N} & F_1 (u) + F_2 (\Psi_c u) + F_3 (\Psi_t u) + F_4 (u) + F_5 (\Phi u). \quad (32)
\end{align*}
\]

The results of the first experiment (deburring problem) on
the test image ‘Mandrill’ are shown in Fig. 2. Figure 2(b)
shows the blurred and noise-added image \( v \) (input). Figure
2(c) is the recovered image by using D-SET recovery. Sharp
dges and detailed textures are seen to be well recovered.
Figures 2(d)–(f) show the pseudo SET components. It is clear
that all the components are well reconstructed. However, we
also remark that the pseudo SET components do not always
match standard intuition perfectly due to the dependency
between edge and texture, for example, in what appears in
the body hair of ‘Mandrill’.

The results from the second experiment (incomplete
measurement recovery problem) on the test image ‘Lena’
are shown in Fig. 3. Figure 3(b) shows the reconstructed
image \( \Phi^* v \) with the Moore-Penrose pseudo inverse of \( \Phi \).
In this case, unlike the deburring case shown in Fig. 2, it is
difficult to visualize the input \( v \) because \( v \) is randomly sampled
noiselet coefficients and is meaningless to human eyes.
Serious artifacts occur in \( \Phi^* v \) due to the singularity of the
measurement matrix \( \Phi \). On the other hand, Fig. 3(c), which
is the restored image by using D-SET recovery, well approxi-
mates the original image in terms of the perceptual quality.
Table 1  Comparison of PSNR [dB] and SSIM in image deblurring.

<table>
<thead>
<tr>
<th></th>
<th>Barbara</th>
<th>Bridge</th>
<th>Building</th>
<th>Cameraman</th>
<th>Lena</th>
<th>Lighthouse</th>
<th>Mandrill</th>
<th>Woman</th>
</tr>
</thead>
<tbody>
<tr>
<td>D-GT recovery [7]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PSNR</td>
<td>29.54</td>
<td>26.64</td>
<td>32.56</td>
<td>30.32</td>
<td>32.16</td>
<td>28.29</td>
<td>28.27</td>
<td>32.06</td>
</tr>
<tr>
<td>SSIM</td>
<td>0.9053</td>
<td>0.8524</td>
<td>0.9323</td>
<td>0.9128</td>
<td>0.9277</td>
<td>0.9024</td>
<td>0.8500</td>
<td>0.9140</td>
</tr>
<tr>
<td>ND-SET recovery</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PSNR</td>
<td>27.09</td>
<td>24.44</td>
<td>30.20</td>
<td>32.36</td>
<td>31.12</td>
<td>28.51</td>
<td>28.11</td>
<td>30.78</td>
</tr>
<tr>
<td>SSIM</td>
<td>0.8424</td>
<td>0.7288</td>
<td>0.8826</td>
<td>0.9089</td>
<td>0.9171</td>
<td>0.8405</td>
<td>0.7270</td>
<td>0.8910</td>
</tr>
<tr>
<td>D-SET recovery (proposed)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PSNR</td>
<td>32.00</td>
<td>27.90</td>
<td>35.30</td>
<td>33.62</td>
<td>34.68</td>
<td>31.02</td>
<td>29.63</td>
<td>34.38</td>
</tr>
<tr>
<td>SSIM</td>
<td>0.9423</td>
<td>0.8966</td>
<td>0.9651</td>
<td>0.9473</td>
<td>0.9552</td>
<td>0.9473</td>
<td>0.8877</td>
<td>0.9517</td>
</tr>
</tbody>
</table>

Table 2  Comparison of PSNR [dB] and SSIM in incomplete measurement recovery.

<table>
<thead>
<tr>
<th></th>
<th>Barbara</th>
<th>Bridge</th>
<th>Building</th>
<th>Cameraman</th>
<th>Lena</th>
<th>Lighthouse</th>
<th>Mandrill</th>
<th>Woman</th>
</tr>
</thead>
<tbody>
<tr>
<td>D-GT recovery [7]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PSNR</td>
<td>29.38</td>
<td>25.88</td>
<td>32.26</td>
<td>36.66</td>
<td>36.93</td>
<td>31.19</td>
<td>28.07</td>
<td>35.48</td>
</tr>
<tr>
<td>SSIM</td>
<td>0.8669</td>
<td>0.8005</td>
<td>0.9162</td>
<td>0.9416</td>
<td>0.9530</td>
<td>0.8813</td>
<td>0.8060</td>
<td>0.9140</td>
</tr>
<tr>
<td>ND-SET recovery</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PSNR</td>
<td>31.92</td>
<td>25.78</td>
<td>32.72</td>
<td>35.64</td>
<td>37.44</td>
<td>30.45</td>
<td>25.52</td>
<td>36.21</td>
</tr>
<tr>
<td>SSIM</td>
<td>0.9241</td>
<td>0.7999</td>
<td>0.9266</td>
<td>0.9304</td>
<td>0.9572</td>
<td>0.8748</td>
<td>0.8210</td>
<td>0.9461</td>
</tr>
<tr>
<td>D-SET recovery (proposed)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PSNR</td>
<td>33.95</td>
<td>26.06</td>
<td>33.07</td>
<td>37.05</td>
<td>38.01</td>
<td>31.60</td>
<td>28.67</td>
<td>36.56</td>
</tr>
<tr>
<td>SSIM</td>
<td>0.9418</td>
<td>0.8041</td>
<td>0.9312</td>
<td>0.9470</td>
<td>0.9615</td>
<td>0.8958</td>
<td>0.8213</td>
<td>0.9478</td>
</tr>
</tbody>
</table>

Fig. 1  8 test images used in the numerical experiments.

Fig. 2  Deblurring on the test image ‘Mandrill’.

Fig. 3  Incomplete measurement recovery on the test image ‘Lena’.
The acceleration of the proposed method by using parallel implementation is our future work.

Acknowledgement

We are grateful to Wemer Wee for his fruitful comments on this paper. This work was partially supported by Grant-in-Aid for JSPS Fellows (24.2522).

References


Appendix A: Elements of Convex Analysis

We list minimum notations in convex analysis. Let $\mathcal{H}$ be a Hilbert space equipped with an inner product $\langle \cdot , \cdot \rangle$ and its induced norm $\| \cdot \|$.

Definition A.1:

(a) (Convex set) A set $C \subseteq \mathcal{H}$ is called convex if $\lambda x + (1 - \lambda) y \in C$ for every $x, y \in C$ and every $\lambda \in [0, 1]$. If a set $C \subseteq \mathcal{H}$ is closed as well as convex, it is called closed convex.

(b) (Proper convex function) A function $f : \mathcal{H} \to (-\infty, \infty]$ is called convex if for every $x, y \in \mathcal{H}$ and every $\lambda \in (0, 1)$

$$f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda) f(y).$$  \hspace{1cm} (A.1)

In particular, a convex function $f : \mathcal{H} \to (-\infty, \infty]$ is called proper if

$$\text{dom}(f) := \{ x \in \mathcal{H} \mid f(x) < \infty \} \neq \emptyset.$$  \hspace{1cm} (A.2)

(c) (Lower semicontinuous function) A function $f : \mathcal{H} \to (-\infty, \infty]$ is called lower semicontinuous if the set $\text{lev}_a(f) := \{ x \in \mathcal{H} \mid f(x) \leq a \}$ is closed for every $a \in \mathbb{R}$ (Note that if $f$ is continuous over $\mathcal{H}$, $f$ is lower semicontinuous). The set of all proper lower semicontinuous convex functions is denoted by $\Gamma_0(\mathcal{H})$.

(d) (Coercivity) A function $f \in \Gamma_0(\mathcal{H})$ is called coercive if

$$\lim_{\|x\| \to \infty} f(x) = \infty.$$  \hspace{1cm} (A.3)

In this case, the existence of a minimizer of $f$, i.e.,

$$\{ x^* \in \mathcal{H} \mid f(x^*) \leq f(x) \} \neq \emptyset$$  \hspace{1cm} (A.4)

is guaranteed.

Appendix B: Proof of Proposition 3.1

Clearly, $F$ is convex and lower semicontinuous on $X$. Moreover, for $(u_1^k, u_2^k, u_3^k) \in X$ s.t. $u_1^k + u_2^k + u_3^k = u_{org}$. Therefore, $F \in \Gamma_0(X)$.

Now it is enough to show the function $F$ is coercive (see Definition A.1(d)). We prove the coercivity of $F$ by contradiction. Suppose there exists a sequence $(u_1^k, u_2^k, u_3^k) \in X$ $(k = 1, 2, 3, \ldots)$ satisfying

$$\lim_{k \to \infty} \| u_1^k, u_2^k, u_3^k \| \to \infty.$$  \hspace{1cm} (A.5)

In this case, there exists $M > 0$ which satisfies

$$\forall p \in \mathbb{N}, \exists k \geq p \quad F(u_1^k, u_2^k, u_3^k) \leq M.$$  \hspace{1cm} (A.7)

which ensures of existence of some subsequence $(u_1^{m_l}, u_2^{m_l}, u_3^{m_l}) \in X$ $(l = 1, 2, 3, \ldots)$ satisfying

$$F(u_1^{m_l}, u_2^{m_l}, u_3^{m_l}) = \| u_1^{m_l} \|_{TV} + \| u_2^{m_l} \|_{TV} + \| u_3^{m_l} \|_{TV}$$

$$+ C(u_1^{m_l} + u_2^{m_l} + u_3^{m_l}) + \| u_1^{m_l} \|_{TV} + \| u_2^{m_l} \|_{TV} + \| u_3^{m_l} \|_{TV}) \leq M \forall l \in \mathbb{N}.$$  \hspace{1cm} (A.8)

By (A.8), we have

$$\| u_1^{m_l} \|_{TV} \leq M \forall l \in \mathbb{N},$$  \hspace{1cm} (A.9)

$$\| u_2^{m_l} \|_{TV} \leq M \forall l \in \mathbb{N},$$  \hspace{1cm} (A.10)

$$\| u_3^{m_l} \|_{TV} \leq M \forall l \in \mathbb{N}.$$  \hspace{1cm} (A.11)

Applying the equivalence of all norms in $\mathbb{R}^N$ and $\Psi_t^c \Psi_t = \Psi_t^c$, $F = F_{TV}$ to (A.9) and (A.10), we have for some $\mu_2, \mu_3 > 0$, that

$$\| u_2^{m_l} \|_2 \leq \| u_2^{m_l} \|_2 \leq \mu_2 M \forall l \in \mathbb{N},$$  \hspace{1cm} (A.12)

$$\| u_3^{m_l} \|_2 \leq \| u_3^{m_l} \|_2 \leq \mu_3 M \forall l \in \mathbb{N}.$$  \hspace{1cm} (A.13)

Moreover, by (A.11)
\[ u_1^{m(l)} + u_2^{m(l)} + u_3^{m(l)} \in C \quad (\forall l \in \mathbb{N}). \]  

(A-14)

Boundedness of \( C \) and (A-12)–(A-14) guarantee the existence of \( \mu_1 > 0 \), s.t.

\[ \|u_1^{m(l)}\|_2 \leq \mu_1 M \quad (\forall l \in \mathbb{N}). \]  

(A-15)

Finally, by (A-12), (A-13), and (A-15), we have

\[ \|(u_1^{m(l)}, u_2^{m(l)}, u_3^{m(l)})\|_X \leq \sqrt{\mu_1^2 + \mu_2^2 + \mu_3^2} M \quad (\forall l \in \mathbb{N}), \]  

(A-16)

which contradicts (A-5).

\[ \square \]

Appendix C: Calculation of the Minimizer of (22)

(22) is obviously equal to

\[ \begin{bmatrix} u_1^{(k+1)} \\ u_2^{(k+1)} \\ u_3^{(k+1)} \end{bmatrix} = \arg\min_{u_1, u_2, u_3} \left( \|z_1 - u_1 - b_1\|^2_2 + \|z_2 - \Psi'(u_1) + b_2\|^2_2 + \|z_3 - \Phi(u_1) + u_2 + b_3\|^2_2 \right), \]  

(A-17)

The solution of (A-17) is obtained by solving the system of the linear equations and we get

\[ u_1^{(k+1)} = (4I_N + 3\Phi'\Phi)^{-1}r, \]  

(A-18)

\[ u_2^{(k+1)} = u_1^{(k+1)} - (z_1 - b_1) + \Psi'(z_2 - b_2), \]  

(A-19)

\[ u_3^{(k+1)} = u_1^{(k+1)} - (z_1 - b_1) + \Phi'(z_3 - b_3), \]  

(A-20)

where

\[ r = (3I_N + 2\Phi'\Phi)(z_1 - b_1) - (I_N + \Phi'\Phi)(\Psi'(z_2 - b_2) + \Psi'(z_3 - b_3)) + (z_4 - b_4) + \Phi'(z_5 - b_5). \]  

(A-21)

Shunsuke Ono received the B.E. and M.E. degree from the Tokyo Institute of Technology in 2010 and 2012, respectively. He joined the Department of Communications and Integrated Systems at the Tokyo Institute of Technology in 2010 and is currently undertaking a Ph.D. course at Tokyo Institute of Technology. His research interests include image processing and optimization.

Takamichi Miyata received the B.E. and M.E. degrees from the University of Toyama in 2001 and 2003, respectively, and Ph.D. degree from Tokyo Institute of Technology in 2006 and joined Department of Communications and Integrated Systems at Tokyo Institute of Technology as an assistant professor. Currently, he is an associate professor in the Department of Electrical, Electronics and Computer Engineering, Chiba Institute of Technology. His current research interests include image processing and image coding.

Isao Yamada is a Professor in the Department of Communications and Integrated Systems, Tokyo Institute of Technology. He received the B.E. degree in computer science from the University of Tsukuba in 1985 and the M.E. and Ph.D. degrees in electrical and electronic engineering from the Tokyo Institute of Technology in 1987 and 1990, respectively. His current research interests are in mathematical signal processing, nonlinear inverse problem and optimization theory. He received Excellent Paper Awards in 1991, 1995, 2006, 2009, the Young Researcher Award in 1992 and the Achievement Award in 2009 from the IEICE; the ICF Research Award from the International Communications Foundation in 2004; the Docomo Mobile Science Award (Fundamental Science Division) from Mobile Communication Fund in 2005; and the Fujino Prize from Tejima Foundation in 2008.

Katsunori Yamaoka received the B.E., M.E., and Ph.D. degrees from the Tokyo Institute of Technology in 1991, 1993 and 2000, respectively. He left Ph.D. program in 1994 and joined the Tokyo Institute of Technology as an assistant professor at that time. In 2000, he joined the National Institute of Multimedia Education (NIME) in Japan as an associate professor at the Tokyo Institute of Technology. Since 2001, he has been an associate professor at the Tokyo Institute of Technology. He has also been a visiting associate professor of the National Institute of Informatics (NII) in Japan since 2004. His research interests include network QoS control for multimedia communications.