

RESEARCH

Open Access

# Image inpainting based on sparse representations with a perceptual metric

Takahiro Ogawa\* and Miki Haseyama

## Abstract

This paper presents an image inpainting method based on sparse representations optimized with respect to a perceptual metric. In the proposed method, the structural similarity (SSIM) index is utilized as a criterion to optimize the representation performance of image data. Specifically, the proposed method enables the formulation of two important procedures in the sparse representation problem, 'estimation of sparse representation coefficients' and 'update of the dictionary', based on the SSIM index. Then, using the generated dictionary, approximation of target patches including missing areas via the SSIM-based sparse representation becomes feasible. Consequently, image inpainting for which procedures are totally derived from the SSIM index is realized. Experimental results show that the proposed method enables successful inpainting of missing areas.

## 1 Introduction

In the field of image processing, there exist many studies on image restoration/enhancement such as image denoising [1-3], image deblurring [4,5], and image inpainting [6]. Furthermore, it is well known that the performance of these studies has been rapidly improved in recent years [1,2,4]. Missing area reconstruction is one of the most attractive topics for study in the field of image restoration since it has a number of applications. Unnecessary object removal, missing block reconstruction in an error-prone environment in wireless communication, and restoration of corrupted old films are representative applications. Since missing area reconstruction can be used in many applications, it has various names including inpainting, image completion, error concealment, and blotch and scratch removal. In this paper, we use 'inpainting' since this is one of the most common names in this research field.

Many inpainting methods for the above applications have been proposed [7-45]. Most methods are broadly classified into two categories: missing structure reconstruction [7-18] and missing texture reconstruction [21-45]. In addition, there have been proposed several inpainting methods which adopt the combined use of the structure and texture reconstruction approaches [20,42].

Variational image inpainting methods which aim at successful structure component reconstruction have traditionally been studied. Variational image inpainting is performed based on the continuity of the geometrical structure of images. Most variational inpainting methods solve partial differential equations (PDEs). One of the pioneering works was proposed by Masnou et al. [7]. Furthermore, Bertalmio et al. proposed a representative image inpainting technique which is based on PDEs. Not only the above methods but also several improved methods have recently been proposed [12-15]. Although these variational image inpainting methods enable successful reconstruction of the structure components, images also include other different important components, i.e., texture components, and alternative methods tend to output better results. The remainder of this paper focuses on the reconstruction of textures with discussion of its details.

Results of pioneering work based on texture synthesis were reported by Efros et al. [21]. Their method is based on the Markov random field model, and inpainting is realized by copying known pixels within a target image. It is well known that successful inpainting of pure texture images can be realized using their method. In recent years, their ideas have been improved by many researchers [22-30].

Drori et al. [23] and Criminisi et al. [24] developed more accurate inpainting techniques. Drori et al. proposed a fragment-based image completion algorithm that

\*Correspondence: ogawa@md.ist.hokudai.ac.jp  
Graduate School of Information Science and Technology, Hokkaido University,  
Sapporo, 060-0814, Japan

can preserve not only textures but also structures within target images. Criminisi et al. proposed an exemplar-based inpainting method, and it became a benchmarking method in this study field. Their method adopts a patch-based greedy sampling algorithm, and faster and simpler inpainting becomes feasible. Recently, many improved versions of the above exemplar-based inpainting method [25–29] have intensively been proposed. Specifically, Meur et al. proposed multiresolution analysis-based inpainting approaches using the exemplar-based method [28,29]. Kwok et al. proposed a much faster inpainting method in which useful schemes for calculating patch similarities in exemplar-based inpainting were introduced [30]. They also reported that their method provided better results than those of the previously reported methods in some cases.

The above existing methods based on texture synthesis and exemplar-based inpainting generally copy pixel values to missing areas directly. Thus, if target images contain uniform and simple textures, the methods can perform accurate inpainting. However, if the above conditions are not satisfied, it becomes difficult to approximate missing textures by only the best matched examples. Therefore, many inpainting methods that approximate patches including missing areas using subspaces generated from known areas within target images have been proposed. In these methods, target patches are generally represented by linear combinations of bases that span the obtained subspaces. The performance of inpainting therefore depends on the generated subspaces and linear coefficients for calculating the linear combination. Amano et al. proposed a principal component analysis (PCA)-based missing area inpainting method using back projection for lost pixels [31]. They utilized an eigenspace that enabled derivation of inverse projection for the inpainting. Several inpainting methods in which kernel methods are introduced into PCA-based subspace construction have also been proposed [32–35]. Based on nonlinear eigenspaces, successful representation of image data becomes feasible, i.e., the methods are suitable for approximating nonlinear structures in images.

Recently, sparse representation for image inpainting has been intensively studied. Sparse representation enables adaptive selection of optimal bases suitable for approximating target images [36,37]. This means subspaces utilized for the inpainting can be adaptively provided. Therefore, several inpainting methods using sparse representation have been proposed [38–42]. Furthermore, Xu et al. have shown the effective use of sparse representation for realizing image inpainting [41]. Specifically, in their method, new modeling of patch priority and patch representation, which are two crucial steps for patch propagation in an exemplar-based inpainting approach, based on sparsity is adopted. In similar ideas,

several inpainting methods based on neighbor embedding approaches are proposed [43,44]. These methods are derived from the aspect of the manifold learning and provide good results. Furthermore, inpainting methods based on rank minimization have also been proposed [45].

The above-described existing methods are based on least squares approximation for inpainting. This means that inpainting minimizing the mean square error (MSE) of intensities, which is the most popular metric, is performed. However, several works [46,47] show that MSE optimal algorithms cannot provide high visual quality. Thus, it may not be appropriate to use MSE as a quality measure for the inpainting. It should be noted that using kernel PCA (KPCA) [32,33], methods such as those shown in [34] and [35] try to approximate nonlinear image features. These methods perform least squares approximation in high-dimensional nonlinear feature spaces, and it has been reported that improvement in performance was achieved in some cases.

Recently, image quality assessment has become popular in overcoming the problem of MSE and its variants. Criteria such as noise quality measure [48], information fidelity criterion [49], and visual information fidelity [50] are well known as perceptual distortion measures, and their performances have been evaluated in detail [51]. The structural similarity (SSIM) index [52] is utilized as one of the most representative quality measures in many fields of image processing. Since its formulation is simple and easy to analyze, the SSIM index can be applied to not only image quality assessment but also design of linear equalizers [53]. Therefore, successful inpainting based on this quality measure can be expected.

In this paper, we present an inpainting method based on sparse representations optimized with respect to a perceptual metric. In order to perform inpainting using sparse representation, the SSIM index is used for a criterion to optimize the representation performance.

Specifically, the proposed method introduces the SSIM-based criterion into two important procedures in the sparse representation problem, i.e., ‘estimation of the sparse representation coefficients’ and ‘update of the dictionary’. This is the biggest difference between the proposed method and existing methods. Then, by deriving the sparse representation of target patches including missing areas based on the generated dictionary, inpainting based on the SSIM index is realized. Note that in the above approach, since optimization problems maximizing the SSIM index are nonconvex, the computation scheme in [53] is adopted, and nonconvex optimization problems are reformulated as quasi-convex problems. In the proposed method, the optimal subspace can be adaptively provided for each target patch using sparse

representation. Furthermore, since the SSIM index, which is a better perceptual criterion than the traditional MSE and its variants, is used, successful inpainting can be expected.

A similar approach has also been proposed by Rehman et al. for realizing noise removal and super-resolution [54]. On the other hand, we present a new scheme for realizing inpainting in this paper, and the target application is different from those in [54]. Basically, in our method, the algorithms for estimation of sparse representation coefficients and generation of the dictionary are different from those in the method of Rehman et al. Furthermore, the biggest difference between our method and the method in [54] is generation of the dictionary. Specifically, in the existing method [54], the dictionary is obtained by directly using the K-SVD algorithm [36], which is based on the MSE-based criterion, where SVD represents singular value decomposition. On the other hand, the proposed method tries to obtain the dictionary based on the SSIM-based criterion, and all of the procedures are based on the SSIM index.

This paper is organized as follows. First, in Section 2, we briefly explain sparse representation and the SSIM index, which are used in the proposed method, as preliminaries. Next, in Section 3, we explain the overview of the proposed method. An inpainting method via sparse representation based on the SSIM index is proposed in Section 4. Experimental results that verify the performance of the proposed method are shown in Section 5. Finally, conclusions are given in Section 6.

## 2 Preliminaries

In this section, we briefly explain sparse representation and the SSIM index used in the proposed method as preliminaries. They are presented in Sections 2.1 and 2.2, respectively.

### 2.1 Sparse representation

Sparse representation of signals is explained in this subsection. The basic algorithm for sparse representation and the K-SVD algorithm [36], which is closely related to the proposed method, are shown in this subsection. Thus, we briefly explain their ideas.

Given an overcomplete dictionary  $\mathbf{D} \in \mathbb{R}^{n \times K}$  whose columns are prototype signal-atoms  $\mathbf{d}_j \in \mathbb{R}^n (j = 1, 2, \dots, K)$ , a target signal  $\mathbf{y} \in \mathbb{R}^n$  can be represented as a sparse linear combination of these atoms<sup>a</sup>. Specifically,  $\mathbf{y}$  is approximated as  $\mathbf{y} \cong \mathbf{D}\mathbf{x} (\mathbf{x} \in \mathbb{R}^K)$ , where  $\mathbf{x}$  is a vector containing the representation coefficients of signal  $\mathbf{y}$ , and it satisfies  $\|\mathbf{y} - \mathbf{D}\mathbf{x}\|_p \leq \epsilon$ . In this subsection, we assume  $p = 2$ .

If  $n < K$  and  $\mathbf{D}$  is a full-rank matrix, an infinite number of solutions are available for the above representation

problem. Thus, a new constraint is introduced into this problem, and the solution is obtained by solving

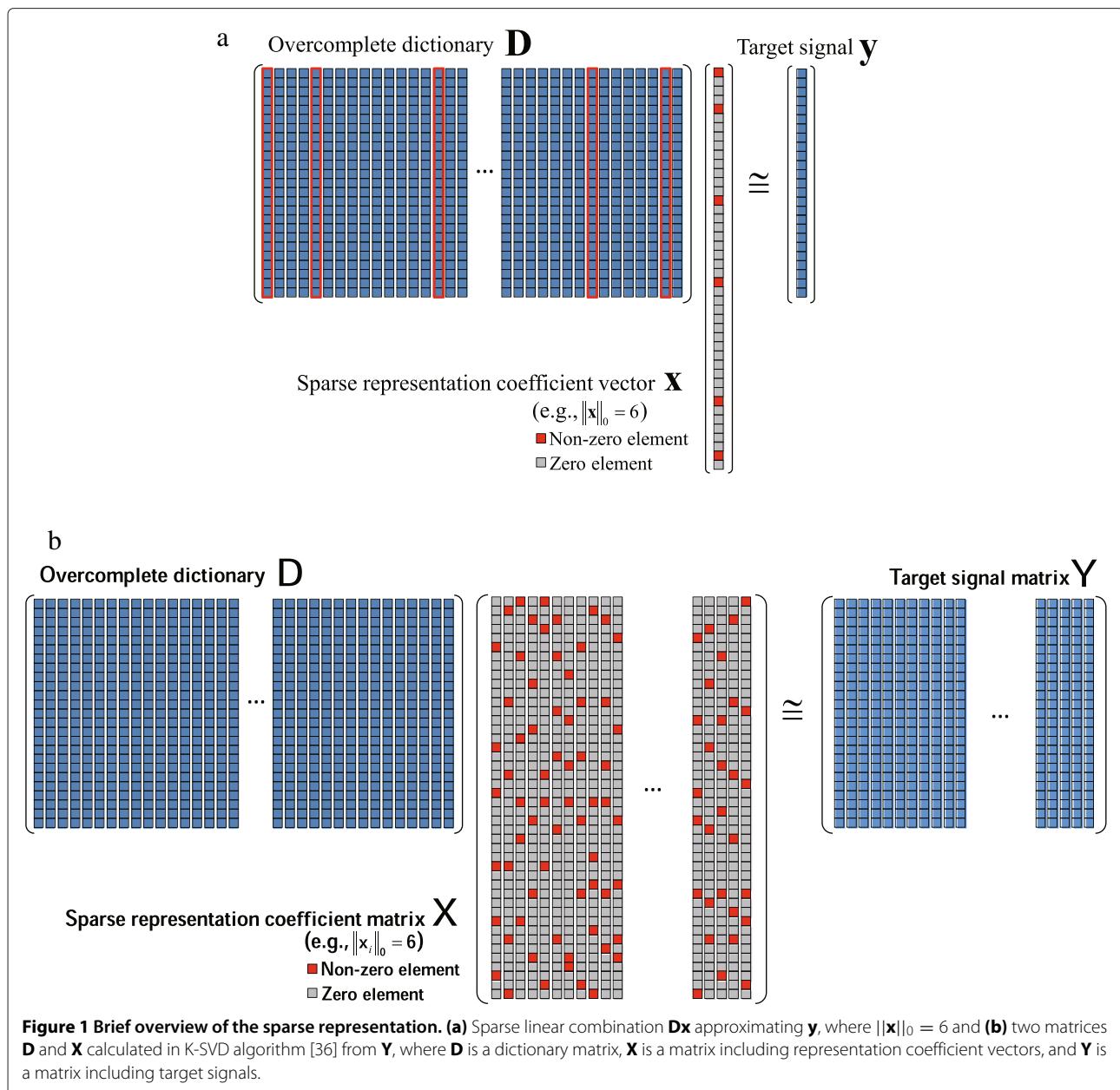
$$\min_{\mathbf{x}} \|\mathbf{y} - \mathbf{D}\mathbf{x}\|_2^2 \quad \text{subject to} \quad \|\mathbf{x}\|_0 \leq T, \quad (1)$$

where  $\|\cdot\|_0$  represents the  $\ell^0$ -norm. Furthermore,  $T$  determines the sparsity of the signals. The above equation represents the optimal representation coefficient vector  $\mathbf{x}$  minimizing the distance  $\|\mathbf{y} - \mathbf{D}\mathbf{x}\|_2^2$  which is calculated under the constraint that the number of the nonzero elements in  $\mathbf{x}$  is  $T$  or less. For example, Figure 1a shows an example of the sparse representation of the target vector  $\mathbf{y}$ , where in this example,  $\|\mathbf{x}\|_0 = 6$ . Therefore, the number of the nonzero elements in  $\mathbf{x}$  is six. By limiting the number of the nonzero elements, we can obtain the solution of the above linear combination. It is well known that calculation of the optimal solution is a nondeterministic polynomial-time hard (NP-hard) problem [55]. Thus, several methods that approximately provide solutions of the above problem have been proposed, and the simplest ones are matching pursuit (MP) [56] and orthogonal MP (OMP) algorithms [57-59]. The basis pursuit algorithm is also a representative algorithm solving the problems by replacing the  $\ell^0$ -norm with an  $\ell^1$ -norm [60]. The focal underdetermined system solver is a similar algorithm using  $\ell^p$ -norm ( $p \leq 1$ ) [61].

Next, given a set of signal vectors  $\mathbf{y}_i (i = 1, 2, \dots, N)$ , there exist dictionary matrices providing the sparse solution  $\mathbf{x}_i$ . The K-SVD algorithm [36] can provide the optimal dictionary matrix  $\mathbf{D}$  and coefficient vectors  $\mathbf{x}_i (i = 1, 2, \dots, N)$  by solving

$$\min_{\mathbf{D}, \mathbf{X}} \{\|\mathbf{Y} - \mathbf{DX}\|_F^2\} \quad \text{subject to} \quad \forall i, \quad \|\mathbf{x}_i\|_0 \leq T, \quad (2)$$

where  $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N]$  and  $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_N]$ , and  $\|\cdot\|_F$  represents the Frobenius norm. In Equation 2, this problem is to obtain the optimal dictionary matrix  $\mathbf{D}$  and representation coefficient vectors  $\mathbf{x}_i (i = 1, 2, \dots, N)$  minimizing the sum of  $\|\mathbf{y}_i - \mathbf{D}\mathbf{x}_i\|^2 (i = 1, 2, \dots, N)$  under the constraint that the number of the nonzero elements in  $\mathbf{x}_i (i = 1, 2, \dots, N)$  is  $T$  or less. Figure 1b shows the relationship between  $\mathbf{Y}$  and  $\mathbf{DX}$ , where the number of the nonzero values in each  $\mathbf{x}_i$  of  $\mathbf{X}$  is six in this example. The K-SVD algorithm approximately calculates the optimal solution of Equation 2 by iterating calculation of  $\mathbf{x}_i (i = 1, 2, \dots, N)$  based on the OMP algorithm and update of the atoms  $\mathbf{d}_j (j = 1, 2, \dots, K)$  in the dictionary matrix  $\mathbf{D}$  using singular value decomposition (SVD). Specifically, the representation coefficient vector  $\mathbf{x}_i (i = 1, 2, \dots, N)$  is estimated one by one, and each atom  $\mathbf{d}_j (j = 1, 2, \dots, K)$  in the dictionary matrix  $\mathbf{D}$  is also updated one by one. As described above, for updating  $\mathbf{d}_j (j = 1, 2, \dots, K)$ , SVD is adopted for effectively providing the approximately optimal solution.



## 2.2 Structural similarity index

The SSIM index represents the similarity between two signal vectors  $\mathbf{y}_1$  and  $\mathbf{y}_2$  ( $\in \mathbb{R}^n$ ), and its specific definition is as follows:

$$\text{SSIM}(\mathbf{y}_1, \mathbf{y}_2) = [l(\mathbf{y}_1, \mathbf{y}_2)]^\alpha \times [c(\mathbf{y}_1, \mathbf{y}_2)]^\beta \times [s(\mathbf{y}_1, \mathbf{y}_2)]^\gamma, \quad (3)$$

where the terms  $l(\mathbf{y}_1, \mathbf{y}_2)$  and  $c(\mathbf{y}_1, \mathbf{y}_2)$  respectively compare the mean and variance of the two signal vectors. Furthermore,  $s(\mathbf{y}_1, \mathbf{y}_2)$  measures their structural correlation. Therefore, from Equation 3, the similarity between two signal vectors is obtained from the three similarities of their luminance, contrast, and structure components, i.e.,

$l(\mathbf{y}_1, \mathbf{y}_2)$ ,  $c(\mathbf{y}_1, \mathbf{y}_2)$ , and  $s(\mathbf{y}_1, \mathbf{y}_2)$ , which are closely related to the human visual system (HVS), where their details are shown below. Note that the parameters  $\alpha > 0$ ,  $\beta > 0$ , and  $\gamma > 0$  determine the relative importance of the three components in Equation 3. Next, the three terms,  $l(\mathbf{y}_1, \mathbf{y}_2)$ ,  $c(\mathbf{y}_1, \mathbf{y}_2)$ , and  $s(\mathbf{y}_1, \mathbf{y}_2)$ , are obtained as

$$l(\mathbf{y}_1, \mathbf{y}_2) = \frac{2\mu_{\mathbf{y}_1}\mu_{\mathbf{y}_2} + C_1}{\mu_{\mathbf{y}_1}^2 + \mu_{\mathbf{y}_2}^2 + C_1}, \quad (4)$$

$$c(\mathbf{y}_1, \mathbf{y}_2) = \frac{2\sigma_{\mathbf{y}_1}\sigma_{\mathbf{y}_2} + C_2}{\sigma_{\mathbf{y}_1}^2 + \sigma_{\mathbf{y}_2}^2 + C_2}, \quad (5)$$

$$s(\mathbf{y}_1, \mathbf{y}_2) = \frac{\sigma_{\mathbf{y}_1, \mathbf{y}_2} + C_3}{\sigma_{\mathbf{y}_1}\sigma_{\mathbf{y}_2} + C_3}. \quad (6)$$

In the above equations,  $\mu_{y_1}$  and  $\mu_{y_2}$  are the means of  $y_1$  and  $y_2$ ,  $\sigma_{y_1}^2$  and  $\sigma_{y_2}^2$  are the variances of  $y_1$  and  $y_2$ , and  $\sigma_{y_1,y_2}$  is the cross covariance between  $y_1$  and  $y_2$ . The constants  $C_1$ ,  $C_2$ , and  $C_3$  are necessary to avoid instability when the denominators are very close to zero.

As shown in [52], the parameters are set as  $\alpha = \beta = \gamma = 1$  and  $C_3 = \frac{C_2}{2}$ , and formulation of the SSIM index is simplified by

$$\text{SSIM}(y_1, y_2) = \frac{(2\mu_{y_1}\mu_{y_2} + C_1)(2\sigma_{y_1,y_2} + C_2)}{\left(\mu_{y_1}^2 + \mu_{y_2}^2 + C_1\right)\left(\sigma_{y_1}^2 + \sigma_{y_2}^2 + C_2\right)}. \quad (7)$$

Note that in the proposed method shown in Section 4,  $C_1 = (K_1 I_{\max})$  and  $C_2 = (K_2 I_{\max})$ , where  $I_{\max} = 255$ ,  $K_1 = 0.01$ , and  $K_2 = 0.03$ . Thus,  $\alpha, \beta, \gamma, C_1, C_2$ , and  $C_3$  are set to the values shown in [52].

In [47] and [52], the effectiveness of the SSIM index as a quality measure, its superiority to MSE, and its variants are presented in detail. Generally, MSE cannot reflect perceptual distortions, and its value becomes higher for images altered with some distortions such as mean luminance shift, contrast stretch, spatial shift, spatial scaling, and rotation but with negligible loss of subjective image quality. Furthermore, blurring severely deteriorates image quality, but its MSE becomes lower than those of the above alterations. On the other hand, the SSIM index is defined by separately calculating three similarities in terms of luminance, variance, and structure, which are derived on the basis of the HVS not accounted for by MSE. Therefore, it becomes a better quality measure providing a solution to the above problem, and this is also confirmed in [47]. We can therefore expect that the use of this similarity for inpainting will provide successful results.

Note that moment invariants take not only image features, such as means and variance, but also image degradations, such as translation, scaling, and rotation, into accounts to generate some invariants and to properly match images without setting any constant. Therefore, in the rest of this subsection, we show some discussions of advantage and disadvantage of the use of the SSIM index by comparing with moment invariants.

### 2.2.1 Advantage

In the proposed method, we use the SSIM index to represent the visual quality of inpainting results. The SSIM index is defined based on several characteristics in the HVS. As shown in Equations 3 to 7, the SSIM index is related to luminance and contrast masking and the correlation. This means that the SSIM index is obtained from the three elements, i.e., Equations 4 to 6. Specifically, the first term defined in Equation 4 is consistent with Weber's

law, which states that the HVS is sensitive to the relative luminance change, and not to the absolute luminance change. The second term defined in Equation 5 is derived based on the contrast masking characteristic that the contrast change is less sensitive when there is a high base contrast than there is a low base contrast. Then, in the third term defined in Equation 6, the structure comparison is conducted after luminance subtraction and contrast normalization. If we ignore  $C_3$ , it is equivalent to calculating the correlation coefficient. In this way, it can be seen that the SSIM index is derived by a bottom-up scheme according to the HVS. This means the proposed method using the SSIM index can perform the inpainting with consideration of the sensitivity to the HVS.

### 2.2.2 Disadvantage

It is known that the SSIM index tends to be robust to translation, scaling, and rotation. However, as those gaps become larger, it also becomes difficult to provide accurate visual quality using the SSIM index due to its definition. On the other hand, moment invariants can output several useful criteria which are invariant under translation, scaling, and rotation. Therefore, if a new visual quality measure can be derived from these moment invariants, successful inpainting based on the derived measure can be also expected. Furthermore, the SSIM index has several parameters compared to the moment invariants.

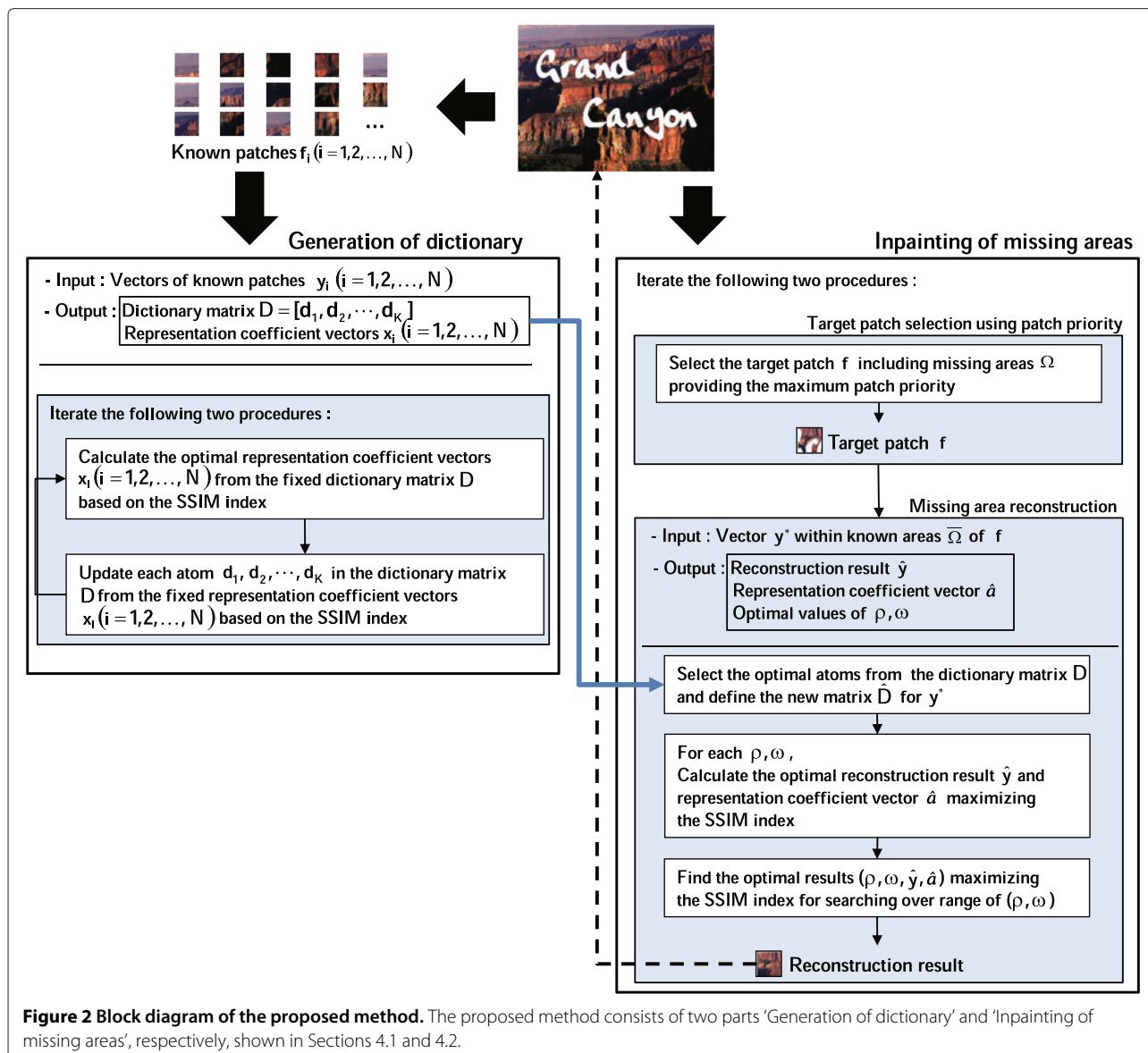
Note that when comparing with the MSE and its variants, the SSIM index can only be calculated from some areas. This means the SSIM index is calculated in a block-wise scheme, not in a pixel-wise scheme. Therefore, to realize the use of the SSIM index for inpainting, we have to adopt the block-wise procedures.

## 3 Overview of our proposed framework

This section presents the overview of the proposed framework. First, we show the outline of the proposed method in Figure 2. As shown in this figure, the proposed method consists of two algorithms, 'generation of dictionary' and 'inpainting of missing areas'. This means these two algorithms respectively correspond to training and test phases.

### 3.1 Generation of dictionary

First, in the generation of the dictionary, we clip known patches not including any missing areas from the target image, and the dictionary matrix  $D$  shown in Section 2.1 is calculated from these patches. In the same manner as the traditional sparse representation problems, we iteratively perform two procedures, 'calculation of the representation coefficients' and 'update of the atoms included in the dictionary matrix  $D$ '. The procedures are similar to those of the traditional method (K-SVD algorithm [36]). The contribution of the proposed method, i.e., the difference



**Figure 2 Block diagram of the proposed method.** The proposed method consists of two parts 'Generation of dictionary' and 'Inpainting of missing areas', respectively, shown in Sections 4.1 and 4.2.

from the traditional method, is the introduction of the SSIM index. Specifically, the representation coefficients and the atoms of the dictionary matrix are calculated in such a way that the SSIM-based approximation performance becomes the highest. This means that the cost function  $\|Y - DX\|_F^2$  in Equation 2 is replaced with that of the SSIM index. Note that in the calculation of the representation coefficients, the maximization problem of the SSIM index is a nonconvex problem, and thus, it is reformulated as a quasi-convex problem using the computation scheme in [53]. On the other hand, in the update of the atoms of the dictionary matrix, we use a simple steepest ascent algorithm since the introduction of the computation scheme in [53] needs high computation

costs. In K-SVD algorithm [36], the atoms can be effectively updated using SVD, but this scheme is based on the least-square approximation, and therefore, we use the simple steepest ascent algorithm.

### 3.2 Inpainting of missing areas

In the inpainting of missing areas, we first clip a patch including missing areas from the target image. Note that we have to determine which patch should be first selected for the inpainting. In the proposed method, we calculate the patch priority for determining the inpainting order based on the method in [24]. Therefore, the patch maximizing the patch priority is selected, and its missing areas are reconstructed in the proposed method.

For the selected patch (denoted as the target patch) including missing areas, the inpainting procedures are performed. Specifically, the proposed method performs the sparse representation of the target patch to estimate the missing intensities. Note that the cost function in Equation 1 is replaced with an SSIM version. Thus, this is the difference from the traditional sparse representation approach and the biggest contribution in our method. The sparse representation of the target patch maximizing the SSIM index is then performed, where this nonconvex maximization problem is also reformulated as a quasi-convex problem using the computation scheme in [53]. In Figure 2, the specific procedures for calculating the sparse representation are shown. Their details are shown in the following section. From the approximation results obtained by the above sparse representation, the proposed method outputs the estimated intensities within the missing areas of the target patch.

By iterating the patch selection based on the patch priority and its SSIM-based missing area reconstruction, we can inpaint the whole missing areas within the target image.

#### 4 Image inpainting via SSIM-based sparse representation

The inpainting method via SSIM-based sparse representation is presented in this section. As described in the previous section, the proposed method is divided into two algorithms, generation of a dictionary and inpainting algorithm. In the first algorithm, the dictionary is generated from known patches  $f_i$  ( $i = 1, 2, \dots, N$ ) within the target image, where  $N$  is the number of known patches, and their size is  $w \times h$  pixels. It should be noted that the proposed method performs calculation of the dictionary based on the new perceptually optimized criterion, i.e., the SSIM index. The details of this calculation are shown in Section 4.1. In the second algorithm, the proposed method clips a patch  $f$  including missing areas from the target image and estimates their unknown intensities. In this algorithm, sparse representation based on the SSIM index is introduced into the inpainting. Its details are shown in Section 4.2. For the following explanation, we denote unknown and known areas within  $f$  as  $\Omega$  and  $\bar{\Omega}$ , respectively.

##### 4.1 Generation of the dictionary

In this subsection, the algorithm for generating the dictionary is presented. In the proposed method, we calculate the dictionary matrix  $\mathbf{D}$  in Equation 2 for reconstructing the missing areas within the target image. Note that the difference from Equation 2 is the use of the SSIM index. In contrast to Equation 2 in minimizing the MSE of the approximation results, the proposed method maximizes the SSIM index of the approximation results by the sparse

representation. Similar to K-SVD algorithm [36], since it is difficult to simultaneously obtain the dictionary matrix and the representation coefficients, we iteratively update these two. Specifically, for the calculation of the representation coefficients optimal in terms of the SSIM index, we use their simple estimation scheme similar to some matching pursuit algorithms. Furthermore, its nonconvex optimization problem is reformulated as a quasi-convex problem using the calculation scheme in [53]. On the other hand, each atom of the dictionary matrix is updated one by one by a simple steepest ascent algorithm. The details are shown below.

As described above, known patches  $f_i$  ( $i = 1, 2, \dots, N$ ) with sizes of  $w \times h$  pixels are clipped from the target image in the same interval. This means that the patches  $f_i$  for generating the dictionary are selected from known parts, which are not damaged, of the target image. Next, for each patch  $f_i$ , we define a vector  $\mathbf{y}_i \in \mathbf{R}^{wh}$ , whose elements are its raster-scanned intensities. Using an overcomplete dictionary matrix  $\mathbf{D} \in \mathbf{R}^{wh \times K}$  containing  $K$  prototype atoms  $\mathbf{d}_j \in \mathbf{R}^{wh}$  ( $j = 1, 2, \dots, K$ ), each vector  $\mathbf{y}_i$  is represented as a sparse linear combination of these atoms,  $\mathbf{y}_i \cong \mathbf{D}\mathbf{x}_i$ , where it satisfies  $\text{SSIM}(\mathbf{y}_i, \mathbf{D}\mathbf{x}_i) \geq \eta$  for a fixed value  $\eta$  that corresponds to  $\epsilon$  in the previous section. The vector  $\mathbf{x}_i \in \mathbf{R}^K$  contains the representation coefficients of  $\mathbf{y}_i$ .

If  $wh < K$  and  $\mathbf{D}$  is a full-rank matrix, an infinite number of solutions are available for the representation problems. Therefore, in the same manner as Equation 1, the proposed method adopts the solution of

$$\max_{\mathbf{x}_i} \text{SSIM}(\mathbf{y}_i, \mathbf{D}\mathbf{x}_i) \quad \text{subject to} \quad \|\mathbf{x}_i\|_0 \leq T. \quad (8)$$

This means that the optimal vector of  $\mathbf{x}_i$  is obtained by maximizing the SSIM index between  $\mathbf{y}_i$  and  $\mathbf{D}\mathbf{x}_i$  under the constraint that the number of the nonzero elements in  $\mathbf{x}_i$  is  $T$  or less. The optimal representation coefficients can then be obtained by solving the above equation.

In addition, according to Equation 2 in the K-SVD algorithm [36], the optimal dictionary matrix  $\mathbf{D}$  can be obtained by solving the following maximization problem:

$$\max_{\mathbf{D}, \mathbf{x}_i (i=1,2,\dots,N)} \sum_{i=1}^N \text{SSIM}(\mathbf{y}_i, \mathbf{D}\mathbf{x}_i) \quad \text{subject to} \quad \forall i, \quad (9)$$

$$\|\mathbf{x}_i\|_0 \leq T.$$

This means that we calculate the dictionary matrix  $\mathbf{D}$  maximizing the approximation performance of all  $\mathbf{y}_i$  ( $i = 1, 2, \dots, N$ ) in terms of the SSIM index under the constraint that the number of the nonzero elements in  $\mathbf{x}_i$  ( $i = 1, 2, \dots, N$ ) is  $T$  or less. In the proposed method, the optimal dictionary matrix  $\mathbf{D}$  is estimated using a scheme similar to the K-SVD algorithm [36], where the procedures are based on the SSIM index. Specifically, this scheme is divided into two procedures, calculation of the optimal

vector  $\mathbf{x}_i (i = 1, 2, \dots, N)$  and update of the dictionary matrix  $\mathbf{D}$ , and they are iteratively performed. We show each of the procedures below.

#### 4.1.1 Calculation of the optimal vector $\mathbf{x}^i$

By fixing the dictionary matrix  $\mathbf{D}$ , the optimal vector  $\mathbf{x}_i$  is calculated for each  $\mathbf{y}_i$ . Specifically,  $\mathbf{x}_i$  can be calculated on the basis of Equation 8. In this optimization problem, we select  $T$  optimal atoms that provide the optimal linear combination based on the SSIM index. Therefore, we adopt the simplest algorithm that selects the optimal atoms one by one, and it is similar to several matching pursuit algorithms [56–59]. Specifically, for each  $\mathbf{y}_i (i = 1, 2, \dots, N)$ , we first search one atom which provides its optimal approximation, maximizing the SSIM index. Furthermore, by adding another atom to the previously selected atoms, we calculate their SSIM-based linear combination approximating each  $\mathbf{y}_i$ , and then, the optimal atom maximizing the SSIM index with the previously selected atoms is selected. Then, by iterating this procedure  $T$  times, the  $T$  optimal atoms can be selected for each  $\mathbf{y}_i$ . Therefore, the procedures are quite simple. In each iteration, we simply select one atom in such a way that the linear combination of this atom and the previously selected atoms maximizes the SSIM index for approximating each  $\mathbf{y}_i (i = 1, 2, \dots, N)$ .

The details of the  $t$ th ( $t = 1, 2, \dots, T$ ) optimal atom selection are shown below.

In the  $t$ th optimal atom selection for  $\mathbf{y}_i$ , the following vector is first defined:

$$\begin{aligned}\mathbf{y}_{i,j}^{(t)} &= \left[ \mathbf{D}_i^{(t-1)} \quad \mathbf{d}_j \right] \begin{bmatrix} \mathbf{x}_i^{(t)} \\ x_j \end{bmatrix} \\ &= \mathbf{D}_{i,j}^{(t)} \mathbf{x}_{i,j}^{(t)} \quad (j = 1, 2, \dots, K),\end{aligned}\quad (10)$$

where  $\mathbf{D}_i^{(t-1)}$  is a  $wh \times (t-1)$  matrix containing  $t-1$  atoms previously selected from  $\mathbf{d}_j (j = 1, 2, \dots, K)$  in  $t-1$  iterations. In addition,

$$\mathbf{D}_{i,j}^{(t)} = \left[ \mathbf{D}_i^{(t-1)} \quad \mathbf{d}_j \right], \quad (11)$$

and

$$\mathbf{x}_{i,j}^{(t)} = \begin{bmatrix} \mathbf{x}_i^{(t)} \\ x_j \end{bmatrix} \quad (\in \mathbb{R}^t) \quad (12)$$

is a coefficient vector for calculating  $\mathbf{y}_{i,j}^{(t)}$ . The vector  $\mathbf{x}_i^{(t)}$  contains representation coefficients that respectively correspond to the atoms in  $\mathbf{D}_i^{(t-1)}$ , and  $x_j$  is that corresponding to  $\mathbf{d}_j$ . Here, we show the specific definitions of  $\mathbf{x}_{i,j}^{(t)}$ ,  $\mathbf{y}_{i,j}^{(t)}$ , and  $\mathbf{D}_{i,j}^{(t)}$ . First,  $\mathbf{x}_{i,j}^{(t)}$  is the sparse representation coefficient vector for representing  $\mathbf{y}_i$  with the atom  $\mathbf{d}_j$  selected to be appended at iteration  $t$ , and  $\mathbf{y}_{i,j}^{(t)}$  is the corresponding approximation of  $\mathbf{y}_i$ . Next,  $\mathbf{D}_{i,j}^{(t)}$  is a matrix including  $t-1$  atoms previously selected in  $t-1$  iterations and the atom

$\mathbf{d}_j$  at iteration  $t$  which are used for representing  $\mathbf{y}_i$ . The proposed method estimates the optimal vector  $\hat{\mathbf{y}}_{i,j}^{(t)}$  of  $\mathbf{y}_{i,j}^{(t)}$  ( $j = 1, 2, \dots, K$ ) that provides the optimal representation performance. Then the optimal atom  $\mathbf{d}_j$  is selected to maximize the SSIM index for the representation of  $\mathbf{y}_i$  by itself together with the atoms selected in the previous  $t-1$  iterations.

In order to calculate  $\hat{\mathbf{y}}_{i,j}^{(t)}$ , the optimal coefficient vector  $\hat{\mathbf{x}}_{i,j}^{(t)}$  in the following equation must be estimated:

$$\hat{\mathbf{y}}_{i,j}^{(t)} = \mathbf{D}_{i,j}^{(t)} \hat{\mathbf{x}}_{i,j}^{(t)}. \quad (13)$$

Thus, we have to solve

$$\hat{\mathbf{x}}_{i,j}^{(t)} = \arg \max_{\mathbf{x}_{i,j}^{(t)}} \text{SSIM} \left( \mathbf{y}_i, \mathbf{y}_{i,j}^{(t)} \right), \quad (14)$$

where  $\text{SSIM} \left( \mathbf{y}_i, \mathbf{y}_{i,j}^{(t)} \right)$  is defined as

$$\text{SSIM} \left( \mathbf{y}_i, \mathbf{y}_{i,j}^{(t)} \right) = \left( \frac{2\mu_{\mathbf{y}_i} \mu_{\mathbf{y}_{i,j}^{(t)}} + C_1}{\mu_{\mathbf{y}_i}^2 + \mu_{\mathbf{y}_{i,j}^{(t)}}^2 + C_1} \right) \left( \frac{2\sigma_{\mathbf{y}_i} \sigma_{\mathbf{y}_{i,j}^{(t)}} + C_2}{\sigma_{\mathbf{y}_i}^2 + \sigma_{\mathbf{y}_{i,j}^{(t)}}^2 + C_2} \right) \quad (15)$$

In this equation,  $\mu_{\mathbf{y}_i} (= \frac{1}{wh} \mathbf{1}' \mathbf{y}_i)$  and  $\sigma_{\mathbf{y}_i}^2 (= \frac{1}{wh} ||\mathbf{y}_i - \mu_{\mathbf{y}_i} \mathbf{1}||^2)$  are respectively the mean and variance of  $\mathbf{y}_i$ , where  $\mathbf{1} = [1, 1, \dots, 1]'$  is a  $wh \times 1$  vector, and the vector/matrix transpose is denoted by the superscript ' in this paper. Similarly,  $\mu_{\mathbf{y}_{i,j}^{(t)}}$  and  $\sigma_{\mathbf{y}_{i,j}^{(t)}}^2$  are the mean and variance of  $\mathbf{y}_{i,j}^{(t)}$ , respectively, and are obtained as follows:

$$\begin{aligned}\mu_{\mathbf{y}_{i,j}^{(t)}} &= \frac{1}{wh} \mathbf{1}' \mathbf{y}_{i,j}^{(t)} \\ &= \frac{1}{wh} \mathbf{1}' \mathbf{D}_{i,j}^{(t)} \mathbf{x}_{i,j}^{(t)} \\ &= \mu_{\mathbf{D}_{i,j}^{(t)}} \mathbf{x}_{i,j}^{(t)},\end{aligned}\quad (16)$$

$$\begin{aligned}\sigma_{\mathbf{y}_{i,j}^{(t)}}^2 &= \frac{1}{wh} \left( \mathbf{y}_{i,j}^{(t)} - \mu_{\mathbf{y}_{i,j}^{(t)}} \mathbf{1} \right)' \left( \mathbf{y}_{i,j}^{(t)} - \mu_{\mathbf{y}_{i,j}^{(t)}} \mathbf{1} \right) \\ &= \frac{1}{wh} \left( \mathbf{D}_{i,j}^{(t)} \mathbf{x}_{i,j}^{(t)} - \frac{1}{wh} \mathbf{1} \mathbf{1}' \mathbf{D}_{i,j}^{(t)} \mathbf{x}_{i,j}^{(t)} \right)' \\ &\quad \times \left( \mathbf{D}_{i,j}^{(t)} \mathbf{x}_{i,j}^{(t)} - \frac{1}{wh} \mathbf{1} \mathbf{1}' \mathbf{D}_{i,j}^{(t)} \mathbf{x}_{i,j}^{(t)} \right) \\ &= \frac{1}{wh} \mathbf{x}_{i,j}^{(t)'} \mathbf{D}_{i,j}^{(t)'} \left( \mathbf{I} - \frac{1}{wh} \mathbf{1} \mathbf{1}' \right) \left( \mathbf{I} - \frac{1}{wh} \mathbf{1} \mathbf{1}' \right) \mathbf{D}_{i,j}^{(t)} \mathbf{x}_{i,j}^{(t)} \\ &= \frac{1}{wh} \mathbf{x}_{i,j}^{(t)'} \mathbf{D}_{i,j}^{(t)'} \mathbf{H}' \mathbf{H} \mathbf{D}_{i,j}^{(t)} \mathbf{x}_{i,j}^{(t)} \\ &= \mathbf{x}_{i,j}^{(t)'} \mathbf{K}_{i,j}^{(t)} \mathbf{x}_{i,j}^{(t)},\end{aligned}\quad (17)$$

where

$$\mu_{\mathbf{D}_{i,j}^{(t)}} = \frac{1}{wh} \mathbf{D}_{i,j}^{(t)'} \mathbf{1}. \quad (18)$$

Furthermore,

$$\mathbf{H} = \mathbf{I} - \frac{1}{wh} \mathbf{1} \mathbf{1}'. \quad (19)$$

is a centering matrix, where  $\mathbf{H} = \mathbf{H}'$  and  $\mathbf{H}^2 = \mathbf{H}$  are satisfied, and  $\mathbf{I}$  is the identity matrix. In addition,

$$\mathbf{K}_{i,j}^{(t)} = \frac{1}{wh} \mathbf{D}_{i,j}^{(t)'} \mathbf{H} \mathbf{D}_{i,j}^{(t)}. \quad (20)$$

In Equation 15,  $\sigma_{\mathbf{y}_i \mathbf{y}_{i,j}^{(t)}}$  is the cross covariance between  $\mathbf{y}_i$  and  $\mathbf{y}_{i,j}^{(t)}$  and is defined as

$$\begin{aligned} \sigma_{\mathbf{y}_i \mathbf{y}_{i,j}^{(t)}} &= \frac{1}{wh} (\mathbf{y}_i - \mu_{\mathbf{y}_i} \mathbf{1})' \left( \mathbf{y}_{i,j}^{(t)} - \mu_{\mathbf{y}_{i,j}^{(t)}} \right) \\ &= \frac{1}{wh} \left( \mathbf{y}_i - \frac{1}{wh} \mathbf{1} \mathbf{1}' \mathbf{y}_i \right)' \left( \mathbf{D}_{i,j}^{(t)} \mathbf{x}_{i,j}^{(t)} - \frac{1}{wh} \mathbf{1} \mathbf{1}' \mathbf{D}_{i,j}^{(t)} \mathbf{x}_{i,j}^{(t)} \right) \\ &= \frac{1}{wh} \mathbf{y}_i' \mathbf{H}' \mathbf{H} \mathbf{D}_{i,j}^{(t)} \mathbf{x}_{i,j}^{(t)} \\ &= \mathbf{k}_{i,j}^{(t)'} \mathbf{x}_{i,j}^{(t)}, \end{aligned} \quad (21)$$

where

$$\mathbf{k}_{i,j}^{(t)} = \frac{1}{wh} \mathbf{D}_{i,j}^{(t)'} \mathbf{H} \mathbf{y}_i. \quad (22)$$

Then, Equation 15 is rewritten as

$$\begin{aligned} \text{SSIM}(\mathbf{y}_i, \mathbf{y}_{i,j}^{(t)}) &= \left[ \frac{2\mu_{\mathbf{y}_i} (\mu_{\mathbf{D}_{i,j}^{(t)'} \mathbf{x}_{i,j}^{(t)}} + C_1)}{\mu_{\mathbf{y}_i}^2 + (\mu_{\mathbf{D}_{i,j}^{(t)'} \mathbf{x}_{i,j}^{(t)}})^2 + C_1} \right] \\ &\quad \times \left[ \frac{2\mathbf{k}_{i,j}^{(t)'} \mathbf{x}_{i,j}^{(t)} + C_2}{\sigma_{\mathbf{y}_i}^2 + \mathbf{x}_{i,j}^{(t)'} \mathbf{K}_{i,j}^{(t)} \mathbf{x}_{i,j}^{(t)} + C_2} \right]. \end{aligned} \quad (23)$$

It should be noted that the criterion in Equation 23 is a nonconvex function of  $\mathbf{x}_{i,j}^{(t)}$ , and it is difficult to obtain the global optimal solution. Thus, we introduce the calculation scheme used in [53] into the estimation of the optimal vector  $\hat{\mathbf{x}}_{i,j}^{(t)}$ . Specifically, the nonconvex problem is transformed into a quasi-convex formulation. The main idea of this scheme is shown as follows. By fixing the mean of  $\mathbf{y}_{i,j}^{(t)}$  ( $= \mu_{\mathbf{D}_{i,j}^{(t)'} \mathbf{x}_{i,j}^{(t)}}$ ), we can focus only on the second term in Equation 23. Therefore, the maximization problem can be simplified.

First, we note that the first term in Equation 23 is a function only of  $\mu_{\mathbf{D}_{i,j}^{(t)'} \mathbf{x}_{i,j}^{(t)}}$  ( $= \rho_{i,j}^{(t)}$ ). Thus, Equation 23 can be rewritten as

$$\begin{aligned} \text{SSIM}(\mathbf{y}_i, \mathbf{y}_{i,j}^{(t)}) &= \left[ \frac{2\mu_{\mathbf{y}_i} \rho_{i,j}^{(t)} + C_1}{\mu_{\mathbf{y}_i}^2 + (\rho_{i,j}^{(t)})^2 + C_1} \right] \\ &\quad \times \left[ \frac{2\mathbf{k}_{i,j}^{(t)'} \mathbf{x}_{i,j}^{(t)} + C_2}{\sigma_{\mathbf{y}_i}^2 + \mathbf{x}_{i,j}^{(t)'} \mathbf{K}_{i,j}^{(t)} \mathbf{x}_{i,j}^{(t)} + C_2} \right]. \end{aligned} \quad (24)$$

Therefore, it can be seen that the first term of the above equation can be fixed by fixing  $\rho_{i,j}^{(t)}$  since  $\mu_{\mathbf{y}_i}$  is a constant.

Then, by constraining  $\mu_{\mathbf{D}_{i,j}^{(t)'} \mathbf{x}_{i,j}^{(t)}} = \rho_{i,j}^{(t)}$ , the optimization problem can be simplified to find

$$\begin{aligned} \hat{\mathbf{x}}_{i,j}^{(t)} \left( \rho_{i,j}^{(t)} \right) &= \arg \max_{\mathbf{x}_{i,j}^{(t)} \in \mathbb{R}^t} \left( \frac{2\mathbf{k}_{i,j}^{(t)'} \mathbf{x}_{i,j}^{(t)} + C_2}{\sigma_{\mathbf{y}_i}^2 + \mathbf{x}_{i,j}^{(t)'} \mathbf{K}_{i,j}^{(t)} \mathbf{x}_{i,j}^{(t)} + C_2} \right) \\ \text{subject to } \mu_{\mathbf{D}_{i,j}^{(t)'} \mathbf{x}_{i,j}^{(t)}} &= \rho_{i,j}^{(t)}. \end{aligned} \quad (25)$$

Thus, the cost function becomes more simple, i.e., we can focus only on the second term of the SSIM index under the constraint fixing  $\mu_{\mathbf{D}_{i,j}^{(t)'} \mathbf{x}_{i,j}^{(t)}} = \rho_{i,j}^{(t)}$ .

Therefore, the overall problem is to find the highest SSIM index by searching over a range of  $\rho_{i,j}^{(t)}$ . Furthermore, the above problem can be rewritten as

$\min : \tau$

subject to

$$\left[ \begin{array}{l} \min : \left[ \tau \left( \sigma_{\mathbf{y}_i}^2 + \mathbf{x}_{i,j}^{(t)'} \mathbf{K}_{i,j}^{(t)} \mathbf{x}_{i,j}^{(t)} + C_2 \right) - \left( 2\mathbf{k}_{i,j}^{(t)'} \mathbf{x}_{i,j}^{(t)} + C_2 \right) \right] \geq 0 \\ \text{subject to } \mu_{\mathbf{D}_{i,j}^{(t)'} \mathbf{x}_{i,j}^{(t)}} = \rho_{i,j}^{(t)} \end{array} \right], \quad (26)$$

and, in the proposed method, the following simple Lagrange multiplier approach is utilized for estimating the optimal vector of  $\mathbf{x}_{i,j}^{(t)}$ :

$$\begin{aligned} \nabla_{\mathbf{x}_{i,j}^{(t)}} \left\{ \tau \left( \sigma_{\mathbf{y}_i}^2 + \mathbf{x}_{i,j}^{(t)'} \mathbf{K}_{i,j}^{(t)} \mathbf{x}_{i,j}^{(t)} + C_2 \right) - \left( 2\mathbf{k}_{i,j}^{(t)'} \mathbf{x}_{i,j}^{(t)} + C_2 \right) \right. \\ \left. + \lambda \left( \mu_{\mathbf{D}_{i,j}^{(t)'} \mathbf{x}_{i,j}^{(t)}} - \rho_{i,j}^{(t)} \right) \right\} = 0, \end{aligned} \quad (27)$$

where the first and second terms correspond to the cost function and the third term corresponds to the constraint. The specific derivations of the above equations are shown in the Appendix. We can then estimate the optimal value of  $\tau$  using a standard bisection procedure, and the optimal vectors  $\hat{\mathbf{x}}_{i,j}^{(t)} \left( \rho_{i,j}^{(t)} \right)$  are calculated for several values of  $\rho_{i,j}^{(t)}$  ( $= \mu_{\mathbf{y}_i} - R\delta, \dots, \mu_{\mathbf{y}_i} - 2\delta, \mu_{\mathbf{y}_i} - \delta, \mu_{\mathbf{y}_i}, \mu_{\mathbf{y}_i} + \delta, \mu_{\mathbf{y}_i} + 2\delta, \dots, \mu_{\mathbf{y}_i} + R\delta$ ) to select  $\hat{\mathbf{x}}_{i,j}^{(t)}$  maximizing Equation 15. Note that  $\delta$  is the searching interval, and  $R$  determines the searching range. Their specific values are shown in Section 5.1. The detailed procedures for estimating  $\tau$  in the proposed method are as follows:

- An initial value of  $\tau$  (say  $\tau_0$ ) is determined between zero to one. Furthermore,  $U_\tau = 1.0$  and  $L_\tau = \tau_0$ , where  $U_\tau$  and  $L_\tau$  respectively represent the upper limit and the lower limit of  $\tau$ . In this paper, we set  $\tau_0 = 0.2$ .
- The optimization problem in Equation 28 is solved using  $\tau$ .

(iii) Two criteria  $S_\tau$  and  $D_\tau$  are calculated as

$$S_\tau = \tau \left( \sigma_{\mathbf{y}_i}^2 + \mathbf{x}_{i,j}^{(t)'} \mathbf{K}_{i,j}^{(t)} \mathbf{x}_{i,j}^{(t)} + C_2 \right) - \left( 2\mathbf{k}_{i,j}^{(t)'} \mathbf{x}_{i,j}^{(t)} + C_2 \right),$$

$$D_\tau = U_\tau - L_\tau.$$

(iv) According to the obtained criteria  $S_\tau$  and  $D_\tau$ , the following steps are operated:

- (a) If  $S_\tau \geq 0$  and  $D_\tau < \epsilon$ , the final optimal solution of  $\tau$  is output, where  $\epsilon = 0.05$ .
- (b) If  $S_\tau \geq 0$  but  $D_\tau \geq \epsilon$ ,  $\tau = \frac{U_\tau + L_\tau}{2}$  and  $U_\tau = \tau$ .
- (c) Otherwise,  $\tau = \frac{U_\tau + L_\tau}{2}$  and  $L_\tau = \tau$ .

(v) Procedures (ii) to (iv) are iterated.

#### 4.1.2 Update of dictionary matrix D

From the calculated optimal vectors  $\mathbf{x}_i$  ( $i = 1, 2, \dots, N$ ), the proposed method updates the dictionary matrix  $\mathbf{D}$ . We update each dictionary element, i.e., each atom, one by one in a greedy fashion. Specifically, we choose one atom and update it in such a way that the representation performance, i.e., the sum of the SSIM index, becomes the highest. We perform the update of each atom  $\mathbf{d}_j$  ( $j = 1, 2, \dots, K$ ) by solving the following problem:

$$\max_{\mathbf{d}_j} \sum_{i|\mathbf{x}_i(j) \neq 0} \text{SSIM}(\mathbf{y}_i, \mathbf{x}_i(j)\mathbf{d}_j), \quad (28)$$

where  $\mathbf{x}_i(j)$  is a  $j$ th element of  $\mathbf{x}_i$ .

In the above equation, we try to maximize the approximation performance of  $\mathbf{y}_i$  ( $i = \{1, 2, \dots, N | \mathbf{x}_i(j) \neq 0\}$ ) by  $\mathbf{x}_i(j)\mathbf{d}_j$ , i.e., by the target atom  $\mathbf{d}_j$  and its corresponding representation coefficient  $\mathbf{x}_i(j)$ . Note that it is difficult to maximize Equation 28 in the same way as the calculation of the optimal vector  $\mathbf{x}_i$  ( $i = 1, 2, \dots, N$ ) since the optimization problem is too complex. Thus, using the well-known steepest ascent algorithm, the proposed method updates each atom  $\mathbf{d}_j$  ( $j = 1, 2, \dots, K$ ). Specifically, the proposed method performs an update of the dictionary matrix  $\mathbf{D}$  by the following procedures:

- Step 1. Select one atom  $\mathbf{d}_j$  ( $j = 1, 2, \dots, K$ ).
- Step 2. Update the selected atom  $\mathbf{d}_j$  by iterating the following equation:

$$\mathbf{d}_j \leftarrow \mathbf{d}_j + \zeta \sum_{i|\mathbf{x}_i(j) \neq 0} \frac{\partial \text{SSIM}(\mathbf{y}_i, \mathbf{x}_i(j)\mathbf{d}_j)}{\partial \mathbf{d}_j}, \quad (29)$$

where  $\zeta$  is a fixed small parameter.

- Step 3. Replace the selected atom  $\mathbf{d}_j$  with the vector obtained by step 2. Note that a new dictionary matrix, whose  $j$ th column, i.e.,  $\mathbf{d}_j$ , is only updated, is obtained.
- Step 4. Repeat steps 1 to 3 for all atoms  $\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_K$  within the dictionary matrix  $\mathbf{D}$ .

Using the above procedures, the proposed method can update the dictionary matrix  $\mathbf{D}$ .

Finally, we clarify the relationship between the K-SVD algorithm [36] and our SSIM-based algorithm. First, the biggest difference between the proposed method and the K-SVD algorithm is the use of different quality metrics. The K-SVD algorithm tries to minimize the MSE for performing sparse representation and dictionary generation. On the other hand, the proposed method tries to maximize the SSIM index for them. Specifically, for the calculation of sparse representation coefficients, we adopt an algorithm similar to the OMP algorithm, but the quality measure is the SSIM index, not the MSE. Therefore, representation coefficients are obtained to maximize the SSIM index which is used as the representation performance. Then, the optimal solution is obtained on the basis of the algorithm used in [53], which is quite different from the algorithm based on the MSE. Furthermore, for generation of the dictionary, the proposed method updates each atom, and its scheme is also similar to that of the K-SVD algorithm. However, the proposed method performs the update of each atom in such a way that the sum of the SSIM index becomes highest and, thus, SVD is not used for the calculation. Then, since the update procedure is too complicated, we simply adopt the steepest ascent algorithm in our method.

#### 4.2 Inpainting algorithm

In this subsection, the inpainting algorithm of the missing area  $\Omega$  in the target patch  $f$  based on the SSIM index is presented. In the proposed method, the target patch  $f$  is approximated by a sparse linear combination of the atoms of the dictionary matrix  $\mathbf{D}$  obtained in the previous subsection. In this approach, we introduce the SSIM index as the approximation performance, and then, the optimal reconstruction results maximizing the SSIM index can be obtained. Note that to obtain the optimal sparse linear combination maximizing the SSIM index, we also introduce the calculation scheme in [53]. Note that different from the previous subsection, since we simultaneously estimate the representation coefficients and the missing intensities, the calculation scheme in [53] is extended. Then, the inpainting of the missing area  $\Omega$  within the target patch  $f$  can be realized based on the SSIM index. The details are shown below.

The proposed method tries to estimate the optimal linear combination

$$\hat{\mathbf{y}} = \mathbf{D}\hat{\mathbf{x}} \quad (30)$$

of the unknown vector  $\mathbf{y}$  of  $f$ , where

$$\{\hat{\mathbf{y}}, \hat{\mathbf{x}}\} = \arg \max_{\mathbf{y}, \mathbf{x}} \text{SSIM}(\mathbf{y}, \mathbf{D}\mathbf{x}) \text{ subject to } \mathbf{E}\mathbf{y} = \mathbf{y}^*, \|\mathbf{x}\|_0 \leq T. \quad (31)$$

Note that  $\mathbf{E}$  ( $\in \mathbf{R}^{N_\Omega \times wh}$ ) is a matrix whose diagonal elements are one or zero, and it extracts only known intensities within  $\mathbf{y}$  to obtain  $\mathbf{y}^*$  ( $\in \mathbf{R}^{N_\Omega}$ ), where  $N_\Omega$  is the

number of known pixels in  $f$ . From Equation 31, the proposed method tries to estimate the unknown vector  $\mathbf{y}$  approximated by the linear combination of the atoms in the dictionary matrix  $\mathbf{D}$  under the constraints that the known intensities in  $\bar{\Omega}$  are fixed and the number of the nonzero elements in  $\mathbf{x}$  is  $T$  or less.

Instead of directly calculating the optimal solution in Equation 31, we first perform the selection of the optimal  $T$  atoms used for approximating  $\mathbf{y}$ . Specifically, the proposed method selects  $T$  optimal atoms from  $\mathbf{D}$  by solving the following problem:

$$\hat{\alpha} = \arg \max_{\alpha} \text{SSIM}(\mathbf{y}^*, \mathbf{ED}\alpha) \text{ subject to } \|\alpha\|_0 \leq T, \quad (32)$$

where its solution can be obtained on the basis of the same algorithm as the calculation of the optimal vector  $\mathbf{x}_i$  described in the previous subsection. Then, a matrix  $\hat{\mathbf{D}}$  containing atoms whose corresponding coefficients in  $\hat{\alpha}$  are nonzero values is obtained.

Next, from the obtained matrix  $\hat{\mathbf{D}}$ , Equation 31 is rewritten as

$$\{\hat{\mathbf{y}}, \hat{\mathbf{a}}\} = \arg \max_{\mathbf{y}, \mathbf{a}} \text{SSIM}(\mathbf{y}, \hat{\mathbf{D}}\mathbf{a}) \text{ subject to } \mathbf{E}\mathbf{y} = \mathbf{y}^*. \quad (33)$$

In the above equation,

$$\begin{aligned} \text{SSIM}(\mathbf{y}, \hat{\mathbf{D}}\mathbf{a}) &= \left[ \frac{2\mu_y\mu_{\hat{\mathbf{D}}\mathbf{a}} + C_1}{\mu_y^2 + \mu_{\hat{\mathbf{D}}\mathbf{a}}^2 + C_1} \right] \left[ \frac{2\sigma_{\mathbf{y}, \hat{\mathbf{D}}\mathbf{a}} + C_2}{\sigma_{\mathbf{y}}^2 + \sigma_{\hat{\mathbf{D}}\mathbf{a}}^2 + C_2} \right] \\ &= \left[ \frac{2(\frac{1}{wh}\mathbf{1}'\mathbf{y})(\mu_{\hat{\mathbf{D}}}\mathbf{a}) + C_1}{(\frac{1}{wh}\mathbf{1}'\mathbf{y})^2 + (\mu_{\hat{\mathbf{D}}}\mathbf{a})^2 + C_1} \right] \\ &\quad \times \left[ \frac{2\mathbf{y}'\hat{\mathbf{H}}\mathbf{a} + whC_2}{\mathbf{y}'\mathbf{H}\mathbf{a} + \mathbf{a}'\hat{\mathbf{D}}'\hat{\mathbf{H}}\mathbf{a} + whC_2} \right], \end{aligned} \quad (34)$$

and

$$\mu_{\hat{\mathbf{D}}} = \frac{1}{wh}\hat{\mathbf{D}}'\mathbf{1}. \quad (35)$$

In the proposed method, we estimate  $\hat{\mathbf{y}}$  and  $\hat{\mathbf{a}}$ , maximizing Equation 34 under the constraint  $\mathbf{E}\mathbf{y} = \mathbf{y}^*$  using the computation scheme in [53] in a similar way shown in the previous subsection. Note that we have to estimate the two vectors, and this computation scheme is extended as follows.

Specifically, Equation 34 is a nonconvex function of  $\mathbf{y}$  and  $\mathbf{a}$ , but the first term in Equation 34 is a function only of  $\frac{1}{wh}\mathbf{1}'\mathbf{y}$  ( $= \rho$ ) and  $\mu_{\hat{\mathbf{D}}}\mathbf{a}$  ( $= \omega$ ) and, thus, we rewrite

Equation 33 in the same way as that in the previous subsection.

$$\begin{aligned} \max_{\mathbf{y}, \mathbf{a}} & \left( \frac{2\mathbf{y}'\hat{\mathbf{H}}\mathbf{a} + whC_2}{\mathbf{y}'\mathbf{H}\mathbf{a} + \mathbf{a}'\hat{\mathbf{D}}'\hat{\mathbf{H}}\mathbf{a} + whC_2} \right) \text{ subject to} \\ & \mathbf{E}\mathbf{y} = \mathbf{y}^*, \frac{1}{wh}\mathbf{1}'\mathbf{y} = \rho, \mu_{\hat{\mathbf{D}}}\mathbf{a} = \omega. \end{aligned} \quad (36)$$

By fixing  $\frac{1}{wh}\mathbf{1}'\mathbf{y} = \rho$  and  $\mu_{\hat{\mathbf{D}}}\mathbf{a} = \omega$ , the first term of the SSIM index shown in Equation 34 can be fixed, and the cost function of Equation 33 can be simplified.

Therefore, the overall problem is to find the highest SSIM index by searching over ranges of  $\rho$  and  $\omega$  as shown in Figure 2. Note that their search ranges are set to  $\mu_{\mathbf{y}^*} - R\delta, \dots, \mu_{\mathbf{y}^*} - 2\delta, \mu_{\mathbf{y}^*} - \delta, \mu_{\mathbf{y}^*}, \mu_{\mathbf{y}^*} + \delta, \mu_{\mathbf{y}^*} + 2\delta, \dots, \mu_{\mathbf{y}^*} + R\delta$ , where  $\mu_{\mathbf{y}^*}$  is the mean of  $\mathbf{y}^*$ . Thus, the solution can be obtained in the same manner as that shown in the previous subsection.

Then, the following problem can be obtained:

$$\begin{aligned} \min : & \tau \\ \text{subject to} : & \end{aligned}$$

$$\begin{aligned} \left[ \min : & [\tau (\mathbf{y}'\mathbf{H}\mathbf{y} + \mathbf{a}'\mathbf{K}_1\mathbf{a} + whC_2) - (\mathbf{y}'\mathbf{K}_2\mathbf{a} + whC_2)] \geq 0 \right. \\ \text{subject to} : & \mathbf{E}\mathbf{y} = \mathbf{y}^*, \frac{1}{wh}\mathbf{1}'\mathbf{y} = \rho, \mu_{\hat{\mathbf{D}}}\mathbf{a} = \omega \end{aligned} \quad (37)$$

where

$$\mathbf{K}_1 = \hat{\mathbf{D}}'\hat{\mathbf{H}}\hat{\mathbf{D}}, \quad (38)$$

$$\mathbf{K}_2 = 2\mathbf{H}\hat{\mathbf{D}}. \quad (39)$$

Note that the optimal value of  $\tau$  can be obtained as shown in the previous subsection.

Furthermore, the proposed method adopts the Lagrange multiplier approach to obtain the optimal vectors of  $\mathbf{y}$  and  $\mathbf{a}$  as follows:

$$\begin{aligned} \nabla_{\mathbf{y}, \mathbf{a}} & \left\{ \tau (\mathbf{y}'\mathbf{H}\mathbf{y} + \mathbf{a}'\mathbf{K}_1\mathbf{a} + whC_2) - (\mathbf{y}'\mathbf{K}_2\mathbf{a} + whC_2) + \sum_{k=1}^{N_{\bar{\Omega}}} \lambda_k \right. \\ & \quad \times (\mathbf{v}_k'\mathbf{y} - y_k^*) + \xi_1 \left( \frac{1}{wh}\mathbf{1}'\mathbf{y} - \rho \right) + \xi_2 (\mu_{\hat{\mathbf{D}}}\mathbf{a} - \omega) \left. \right\} = 0, \end{aligned} \quad (40)$$

where  $\mathbf{v}_k (k = 1, 2, \dots, N_{\bar{\Omega}})$  is a vector satisfying

$$\mathbf{E} = [\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_{N_{\bar{\Omega}}}]', \quad (41)$$

and  $y_k^* (k = 1, 2, \dots, N_{\bar{\Omega}})$  satisfies

$$\mathbf{y}^* = [y_1^*, y_2^*, \dots, y_{N_{\bar{\Omega}}}^*]'. \quad (42)$$

In Equation 40, the first and second terms are from the cost function, and the third, fourth, and fifth terms are from the constraints.

Then, by solving the above problem, the proposed method can calculate the optimal vectors  $\hat{\mathbf{a}}$  and  $\hat{\mathbf{y}}$ . Finally,

from the obtained result  $\hat{y}$ , the proposed method outputs the estimated intensities in the missing area  $\Omega$ .

As shown in the above procedures, we can estimate the missing intensities in  $\Omega$  within the target patch  $f$ . Therefore, the proposed method clips patches including missing areas and performs inpainting to estimate all missing intensities. This means that the proposed method gradually reconstructs missing areas patch by patch starting from the missing boundary. It should be noted that in order to realize this scheme, we have to determine the order in which patches along the fill-front  $\partial\Omega$  of missing areas are filled. We call this order ‘patch priority’. In the proposed method, patch priorities are determined by the method proposed by Criminisi et al. [24]. Specifically, given a patch  $f_p$  centered at pixel  $p$  that is in the fill-front of the missing areas within the target image, its priority  $P(p)$  is defined as follows:

$$P(p) = C(p) \cdot D(p), \quad (43)$$

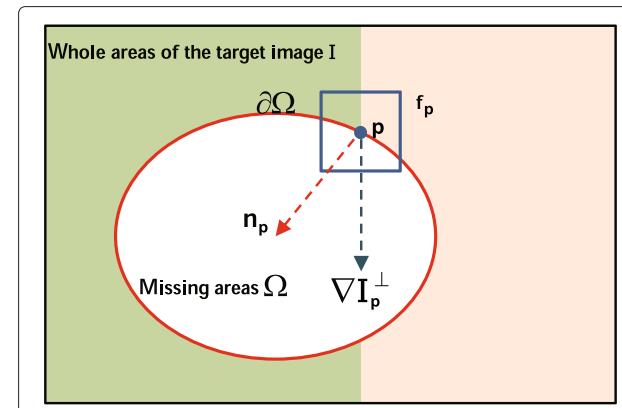
where  $C(p)$  and  $D(p)$  are called confidence term and data term, respectively, and they are defined as follows:

$$C(p) = \frac{\sum_{q \in f_p \cap (I - \Theta)} C(q)}{\text{area}(f_p)}, \quad (44)$$

$$D(p) = \frac{|\nabla I_p^\perp \cdot n_p|}{I_{\max}}. \quad (45)$$

In the above equations,  $I$  and  $\Theta$  are the whole areas of the target image and whole missing areas, respectively. Furthermore,  $\text{area}(f_p)$  ( $= w \times h$ ) represents the number of pixels included within the target patch  $f_p$ . Then,  $I_{\max}$  is a normalization factor (e.g.,  $I_{\max} = 255$  for a typical gray scale image),  $\nabla I_p^\perp$  is an isophote at pixel  $p$ , and  $n_p$  is a unit vector orthogonal to the fill-front at pixel  $p$ . Note that  $C(p)$  is initially set as  $C(p) = 0 \forall p \in \Theta$  and  $C(p) = 1 \forall p \in (I - \Theta)$ . After performing the inpainting,  $C(p)$  is substituted into those in the inpainted areas for the following inpainting process.

Note that the confidence term represents the mean reliability of the pixels within the target patch  $f_p$ . Therefore, if the target patch  $f_p$  contains many known intensities, its value becomes higher. Furthermore, after the inpainting, the reconstructed pixels have the values less than one, i.e., the reconstructed pixels have higher reliability than that of the missing pixels but lower reliability than that of the original pixels. Furthermore, as shown in Figure 3, the data term is a function of the strength of isophotes at the fill front  $\partial\Omega$  [24]. Therefore, by calculating the inner product of the isophote  $\nabla I_p^\perp$  at pixel  $p$  and unit vector  $n_p$  orthogonal to the fill-front at pixel  $p$ , the linear structures can be reconstructed first. In this way, we can restore all of the missing areas within the target image according to the patch priorities in Equation 43.



**Figure 3** Overview of the data term calculation. Given the target patch  $f_p$ ,  $n_p$  is the normal to the contour of the fill front  $\partial\Omega$  of the target missing areas  $\Omega$  and  $\nabla I_p^\perp$  is the isophote at pixel  $p$ .

## 5 Experimental results

In this section, we verify the performance of the proposed method in order to confirm its effectiveness. First, we show results of subjective evaluation of the proposed method using several test images. Furthermore, results of quantitative evaluation using peak signal-to-noise ratio (PSNR) obtained from MSE and the SSIM index are shown, and the effectiveness of the use of the SSIM index is also discussed.

In this section, we show the conditions of the experiments in Section 5.1. In this subsection, we mainly explain the details of the experiments and the comparative methods. In Section 5.2, subjective and quantitative results are shown in comparison with those of the existing methods, and the effectiveness of the proposed method is also discussed. In Section 5.3, we show some examples by applying the proposed method to test images including larger missing areas.

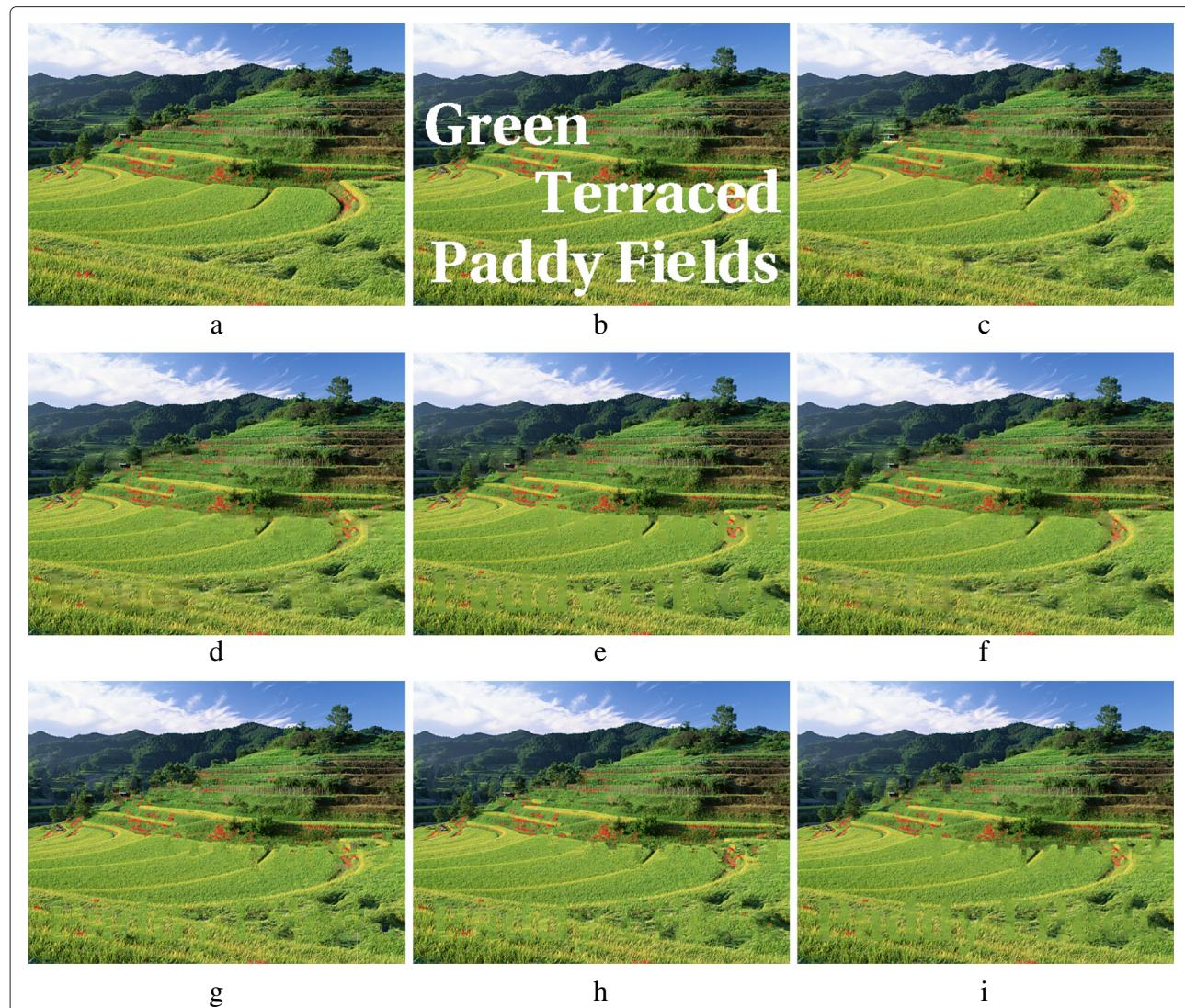
### 5.1 Conditions of experiments

In this subsection, we explain the conditions of the experiments. In the experiments, we first prepared three test images, which are shown in Figures 4, 5, and 6. Furthermore, we added text regions to these test images and obtained corrupted images.

We performed inpainting of the three corrupted test images using the proposed method and the following existing methods:

1. Methods based on PCA or KPCA [31,34,35]

These existing methods generate eigenspaces or nonlinear eigenspaces of patches for inpainting based on PCA or KPCA. Since it is well known that eigenspaces can provide least-square approximation of target data, i.e., eigenspaces are the optimal subspaces based on MSE, the method in [31] is

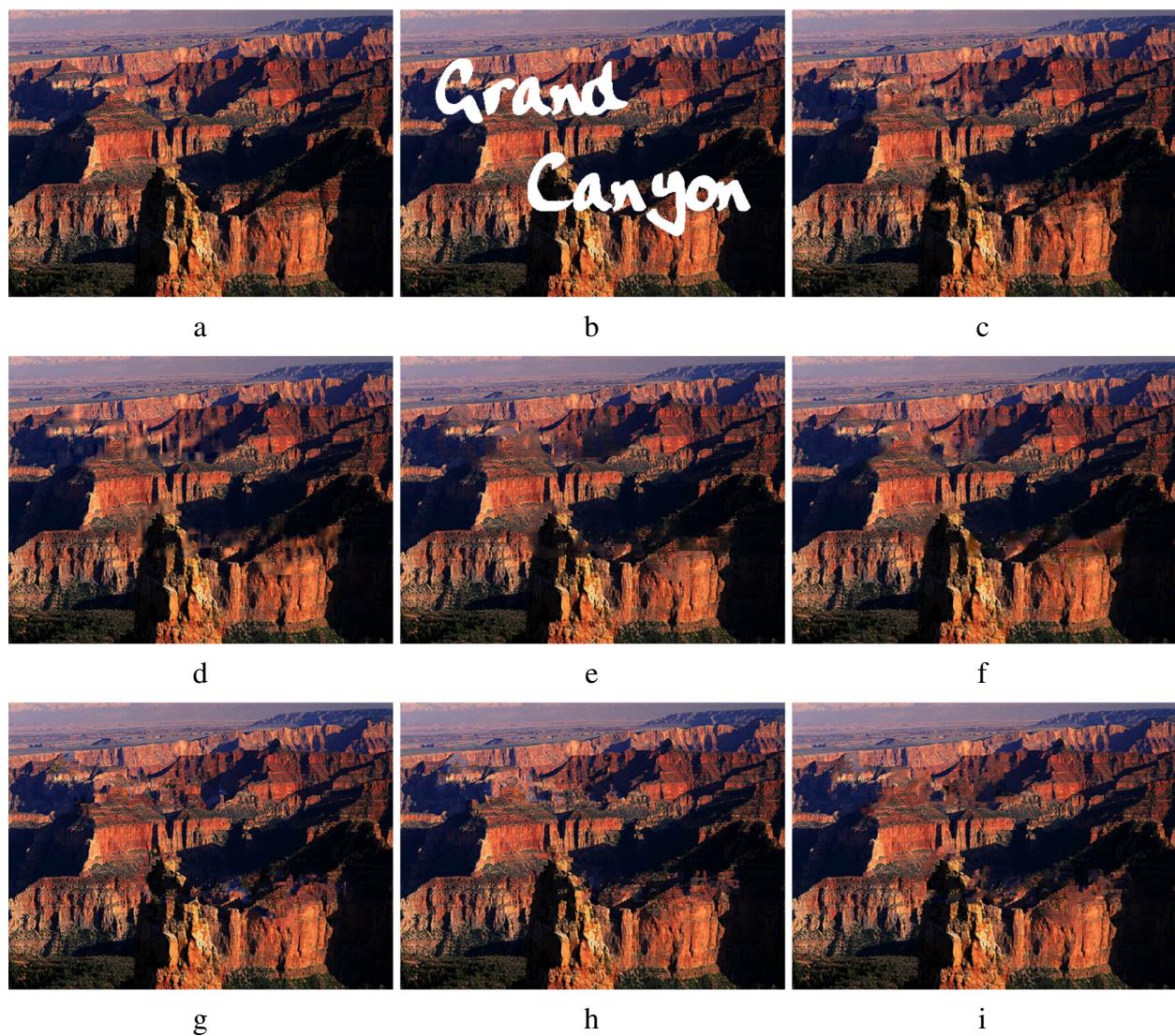


**Figure 4** Obtained results of image 1. **(a)** Original image ( $480 \times 360$  pixels, 24-bit color levels), **(b)** corrupted image including text regions (11.9% loss), **(c)** results obtained by the proposed method, **(d)** results obtained by the method in [31], **(e)** results obtained by the method in [34], **(f)** results obtained by the method in [35], **(g)** results obtained by the method in [24], **(h)** results obtained by the method in [30], and **(i)** results obtained by the method in [41].

- suitable for comparison with the proposed method. Furthermore, the methods in [34] and [35] utilize nonlinear eigenspaces to perform the approximation of nonlinear texture features in images, and we therefore used these methods in the experiments.
2. Exemplar-based inpainting methods [24,30]  
Several exemplar-based inpainting methods have been proposed. The method in [24] is a representative method, and its improved version was proposed in [30], both methods being based on least-square error approaches. In the proposed method, we determine the patch priority using the scheme in [24] and, thus, the difference between our method and [24] is the algorithm for estimating

missing intensities. Therefore, the method in [24] is suitable in confirming the effectiveness of the proposed inpainting algorithm, i.e., the missing intensity estimation algorithm. Furthermore, although the method in [30] improves on the speed rather than inpainting performance improvement, it is reported in their paper that their method improves the performance of [24] in some cases. Therefore, in the experiments, we used these methods as comparative methods.

3. Sparse representation-based inpainting methods [41]  
As described above, the method in [41] adopts the new modeling of patch priority and patch representation, which are two crucial steps for patch



**Figure 5** Obtained results of image 2. (a) Original image ( $480 \times 359$  pixels, 24-bit color levels), (b) corrupted image including text regions (8.9% loss), (c) results obtained by the proposed method, (d) results obtained by the method in [31], (e) results obtained by the method in [34], (f) results obtained by the method in [35], (g) results obtained by the method in [24], (h) results obtained by the method in [30], and (i) results obtained by the method in [41].

propagation in the exemplar-based inpainting approach, based on sparsity. It should be noted that since this method is based on sparse representation but uses MSE-based criteria, it is suitable for comparison.

In this paper, we regard those in [35] and [41] as state-of-the-art methods.

Furthermore, the method in [41] has improved the performance in both patch approximation improvement based on sparse representation and patch priority estimation. Thus, we regard this method as a state-of-the-art method.

In the experiments, we used the above methods as comparative methods for evaluation of our method. For performing inpainting by the proposed method and existing methods [24,30,41], patch size was fixed to 15 ( $w = h = 15$ ). Furthermore, the existing methods in [31,34] and [35] simply perform inpainting in a raster scanning order. Then, for some test images, since target patches contain missing areas in the whole parts, those methods cannot perform inpainting on those missing areas. Thus, in the experiments, patch size was set to 30. Note that much smaller patches were used in some existing methods in previous studies and that accurate performance could be achieved. In these experiments, we used such



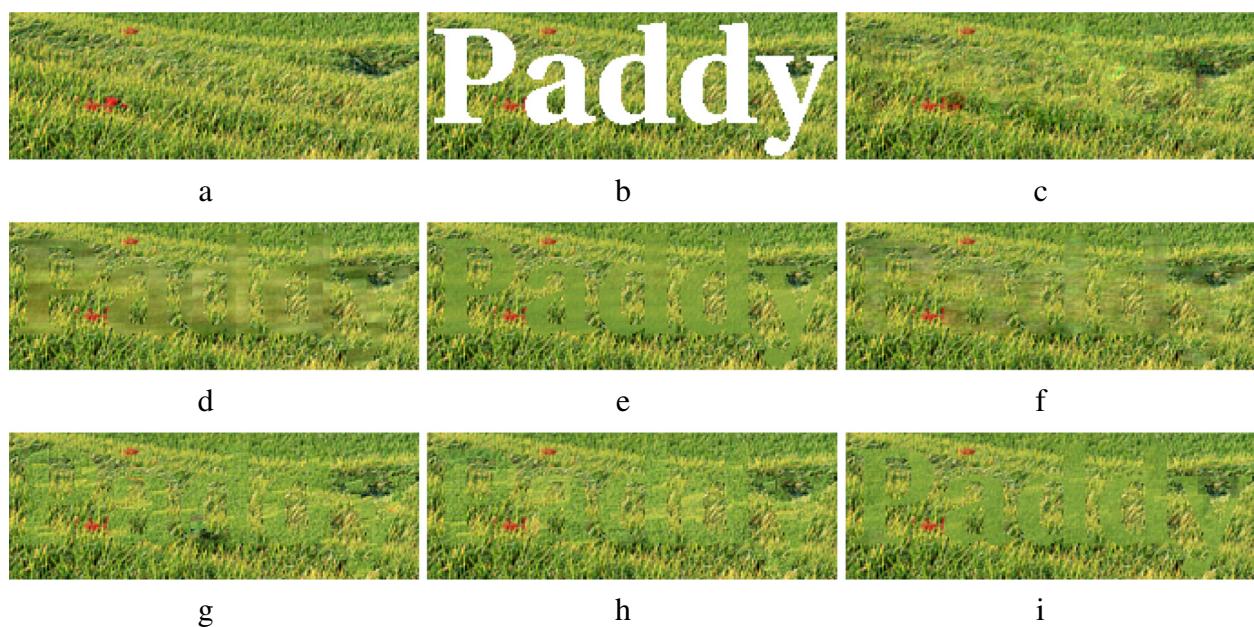
**Figure 6** Obtained results of image 3. **(a)** Original image ( $480 \times 360$  pixels, 24-bit color levels), **(b)** corrupted image including text regions (8.9% loss), **(c)** results obtained by the proposed method, **(d)** results obtained by the method in [31], **(e)** results obtained by the method in [34], **(f)** results obtained by the method in [35], **(g)** results obtained by the method in [24], **(h)** results obtained by the method in [30], and **(i)** results obtained by the method in [41].

difficult conditions in order to make the difference in the performances of the proposed method and the existing methods clearer. Furthermore, in our method, we simply determined  $T = 10$ ,  $\delta = 5$ , and  $R = 6$ .

## 5.2 Subjective and quantitative evaluations

Based on the experimental conditions shown in the previous subsection, inpainting was performed using the proposed method and existing methods. Figures 4, 5, and 6 show the results obtained by those methods. For better subjective evaluation, we also show their zoomed portions in Figures 7, 8, and 9, respectively. From the obtained results, we can confirm that the

proposed method successfully performs inpainting without suffering from oversmoothness. Some MSE criterion-based methods also accurately perform inpainting, but it becomes difficult to simultaneously maintain sharpness in some cases. In some existing methods such as [24], [30], and [41], the performance becomes worse than that reported in those papers. As described above, in this paper, we selected conditions different from those used in those paper, i.e., larger size patches were used. Since this comparison scheme was adopted in several papers, we also used such difficult conditions in order to make the difference in the performance of the proposed method and existing methods clearer. Then, the representation

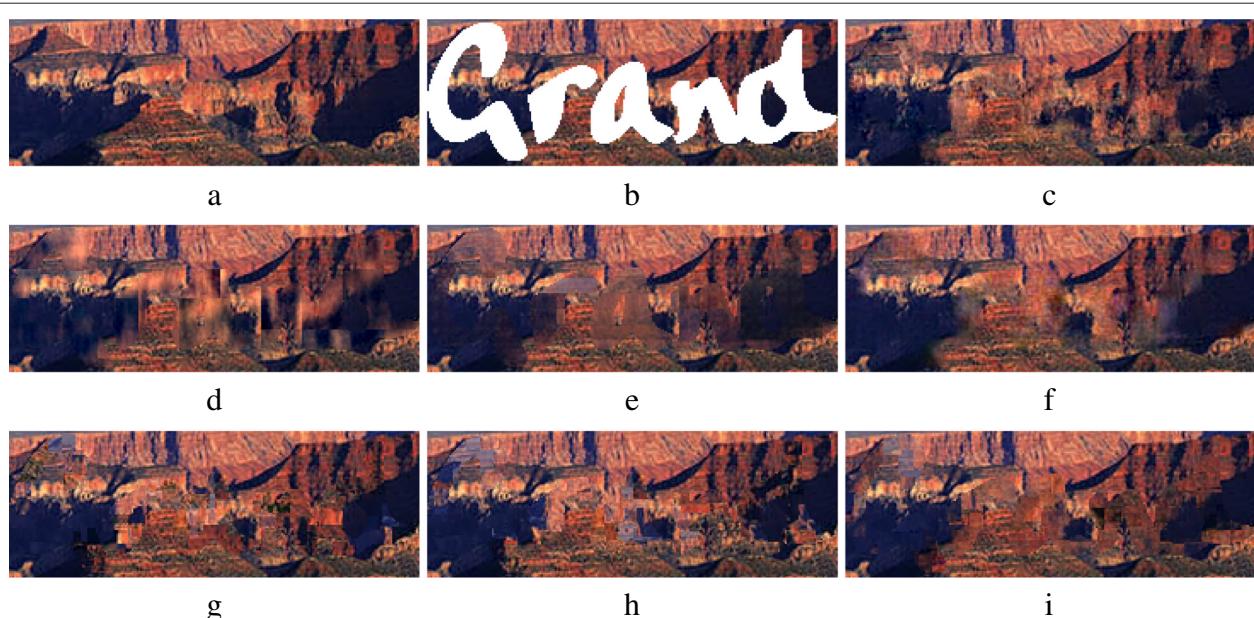


**Figure 7** Zoomed portions of the results shown in Figure 4. (a-i) respectively correspond to the zoomed portions of Figures 4a-i.

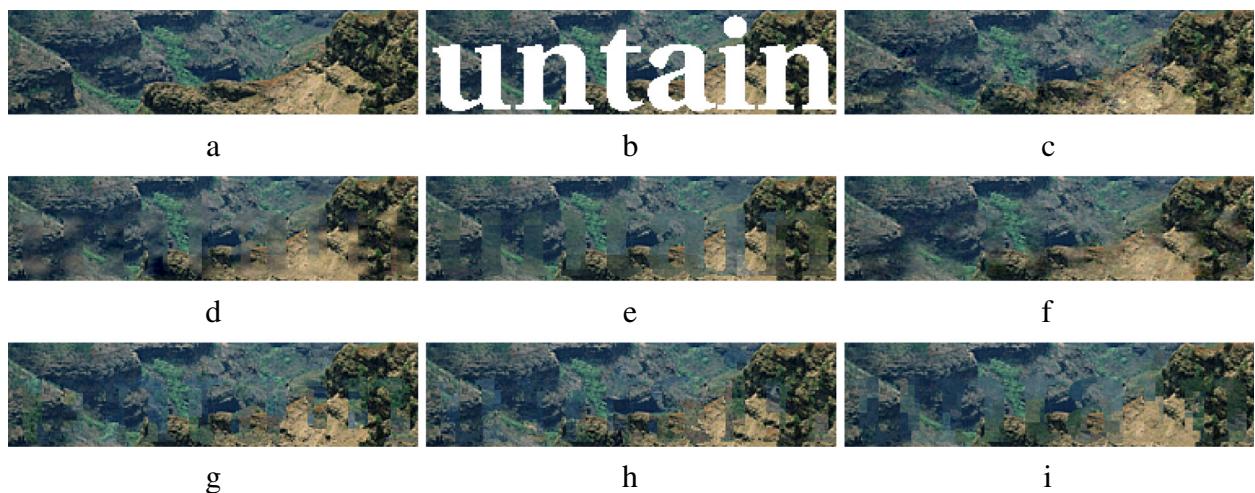
abilities of the methods become worse and the obtained results tend to be blurred. Since the exemplar-based methods in [24] and [30] directly select known patches from the target image for inpainting, blurring tends to be reduced. Nevertheless, even in those methods, it is difficult to perfectly remove degradation. Furthermore, although the methods in [34] and [35] adopt nonlinear eigenspaces for inpainting to represent nonlinear texture features, their

representation abilities become worse as the dimension of the subspace becomes smaller, where the dimension was set to the same as that of the proposed method.

Generally, natural images contain much more powers in low-frequency components than those in high-frequency components. Low-dimensional subspaces obtained from the MSE-based criteria in the existing methods therefore tend to represent only such low-frequency components.



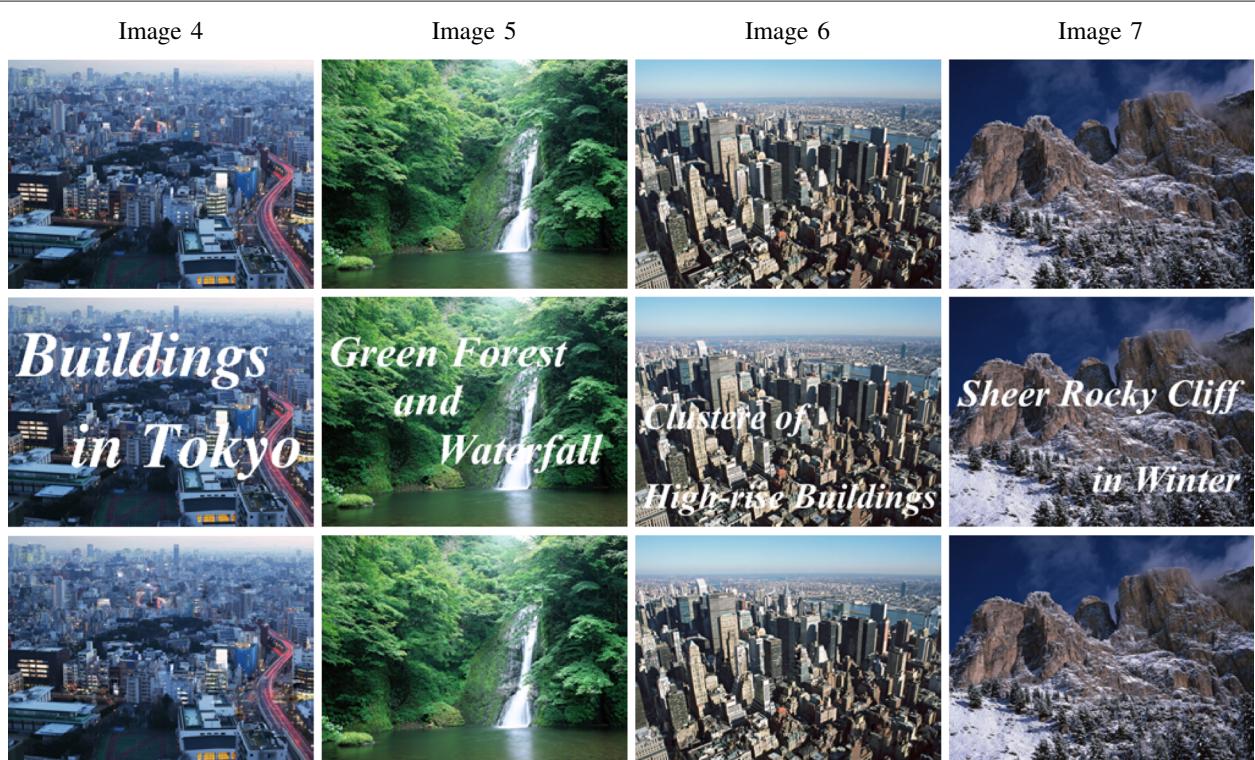
**Figure 8** Zoomed portions of the results shown in Figure 5. (a-i) respectively correspond to the zoomed portions of Figures 5a-i.



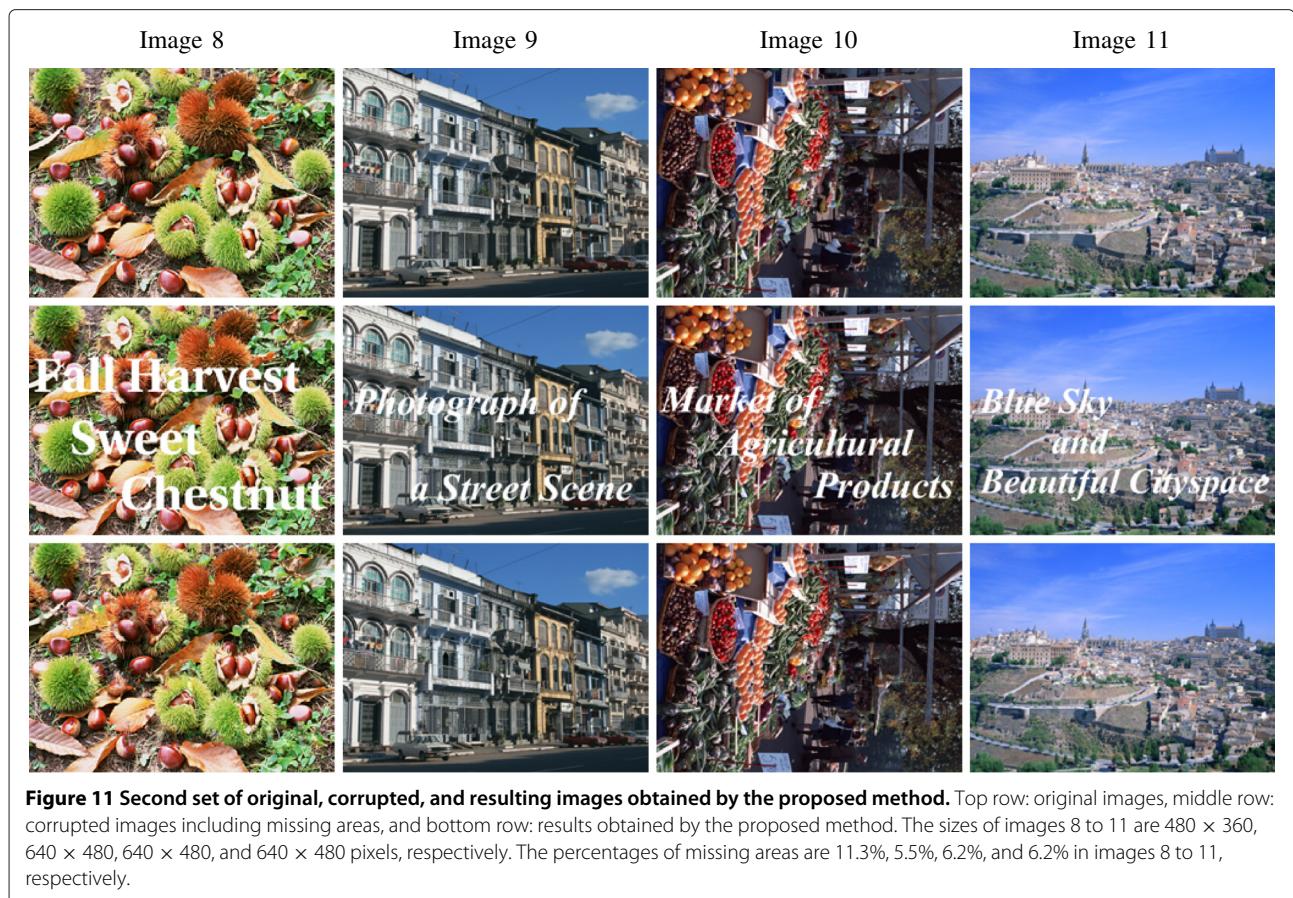
**Figure 9** Zoomed portions of the results shown in Figure 6. (a-i) respectively correspond to the zoomed portions of Figures 6a-i.

Thus, since it becomes difficult to represent high-frequency components, their results suffer from over-smoothness. On the other hand, the SSIM index contains a term comparing components not including average components, i.e., variances, as shown in Equation 5,

and, thus, subspaces used for inpainting tend to successfully represent high-frequency components. Therefore, the proposed method can perform inpainting successfully. Furthermore, the proposed method adopts sparse representation in addition to the SSIM index. This approach



**Figure 10** First set of original, corrupted, and resulting images obtained by the proposed method. Top row: original images, middle row: corrupted images including missing areas, and bottom row: results obtained by the proposed method. The sizes of images 4 to 7 are 480 × 359, 640 × 480, 640 × 480, and 640 × 480 pixels, respectively. The percentages of missing areas are 10.7%, 6.7%, 5.5%, and 5.4% in images 4 to 7, respectively.



enables adaptive selection of the optimal atoms for each target patch including missing areas. This means that the optimal subspace can be provided for each target patch by our method.

Next, we show results of quantitative evaluation for the proposed method and the existing methods. Eight test images shown in Figures 10 and 11 are added to those in

Figures 4, 5, and 6. In these figures, the results of inpainting by our method are also shown. Tables 1 and 2 show the results of PSNR (dB), which is calculated from MSE, and the SSIM index of inpainting results, respectively. Note that since the inpainting is performed for each patch, the SSIM index is calculated for the patches, and their average values are shown in the tables. In addition, the evaluation

**Table 1 Performance comparison (PSNR) of the proposed method and existing methods**

Image number	[31]	[34]	[35]	[24]	[30]	[41]	Proposed method
1 (Figure 4)	18.55	18.13	18.75	16.85	16.72	17.51	17.22
2 (Figure 5)	16.51	15.98	17.68	14.26	14.68	15.08	15.51
3 (Figure 6)	19.93	19.48	20.57	18.01	17.86	18.98	18.62
4 (Figure 10, first column)	15.95	16.51	17.86	14.97	15.19	15.97	15.85
5 (Figure 10, second column)	17.06	16.86	17.81	15.70	15.51	16.08	16.14
6 (Figure 10, third column)	14.42	13.81	15.86	11.79	12.20	12.02	13.38
7 (Figure 10, fourth column)	15.93	16.07	16.51	15.22	15.49	15.27	14.05
8 (Figure 11, first column)	12.98	12.74	14.17	11.34	10.92	11.81	12.04
9 (Figure 11, second column)	15.59	15.57	17.79	13.56	13.55	13.42	15.55
10 (Figure 11, third column)	14.38	14.66	16.56	13.42	13.43	13.82	13.86
11 (Figure 11, fourth column)	16.80	17.09	18.68	15.47	15.48	15.98	16.00

**Table 2 Performance comparison (SSIM) of the proposed method and existing methods**

Image number	[31]	[34]	[35]	[24]	[30]	[41]	Proposed method
Image 1 (Figure 4)	0.6355	0.6090	0.6411	0.6822	0.6773	0.7145	0.7811
Image 2 (Figure 5)	0.5130	0.5154	0.5999	0.5077	0.5277	0.5308	0.6454
Image 3 (Figure 6)	0.6248	0.6051	0.6538	0.7318	0.7246	0.7569	0.8156
Image 4 (Figure 10, first column)	0.5833	0.5762	0.6373	0.6563	0.6708	0.7036	0.7806
Image 5 (Figure 10, second column)	0.6419	0.6424	0.6774	0.7298	0.7196	0.7410	0.8049
Image 6 (Figure 10, third column)	0.6460	0.6458	0.7346	0.6750	0.6933	0.6756	0.7619
Image 7 (Figure 10, fourth column)	0.6711	0.6766	0.7134	0.7478	0.7521	0.7402	0.7722
Image 8 (Figure 11, first column)	0.5871	0.5522	0.6282	0.6561	0.6394	0.6840	0.6940
Image 9 (Figure 11, second column)	0.6501	0.6599	0.7645	0.6852	0.6799	0.6700	0.8072
Image 10 (Figure 11, third column)	0.6240	0.6295	0.7185	0.6992	0.6980	0.7060	0.7681
Image 11 (Figure 11, fourth column)	0.6864	0.7069	0.7685	0.7155	0.7108	0.7352	0.8039

values are computed only on the reconstructed pixels. The results show that several existing methods have higher PSNR values, i.e., lower MSE values, than that of the proposed method. Specifically, the existing method in [35] and the proposed method output the best results in terms of PSNR (MSE) and the SSIM index, respectively.

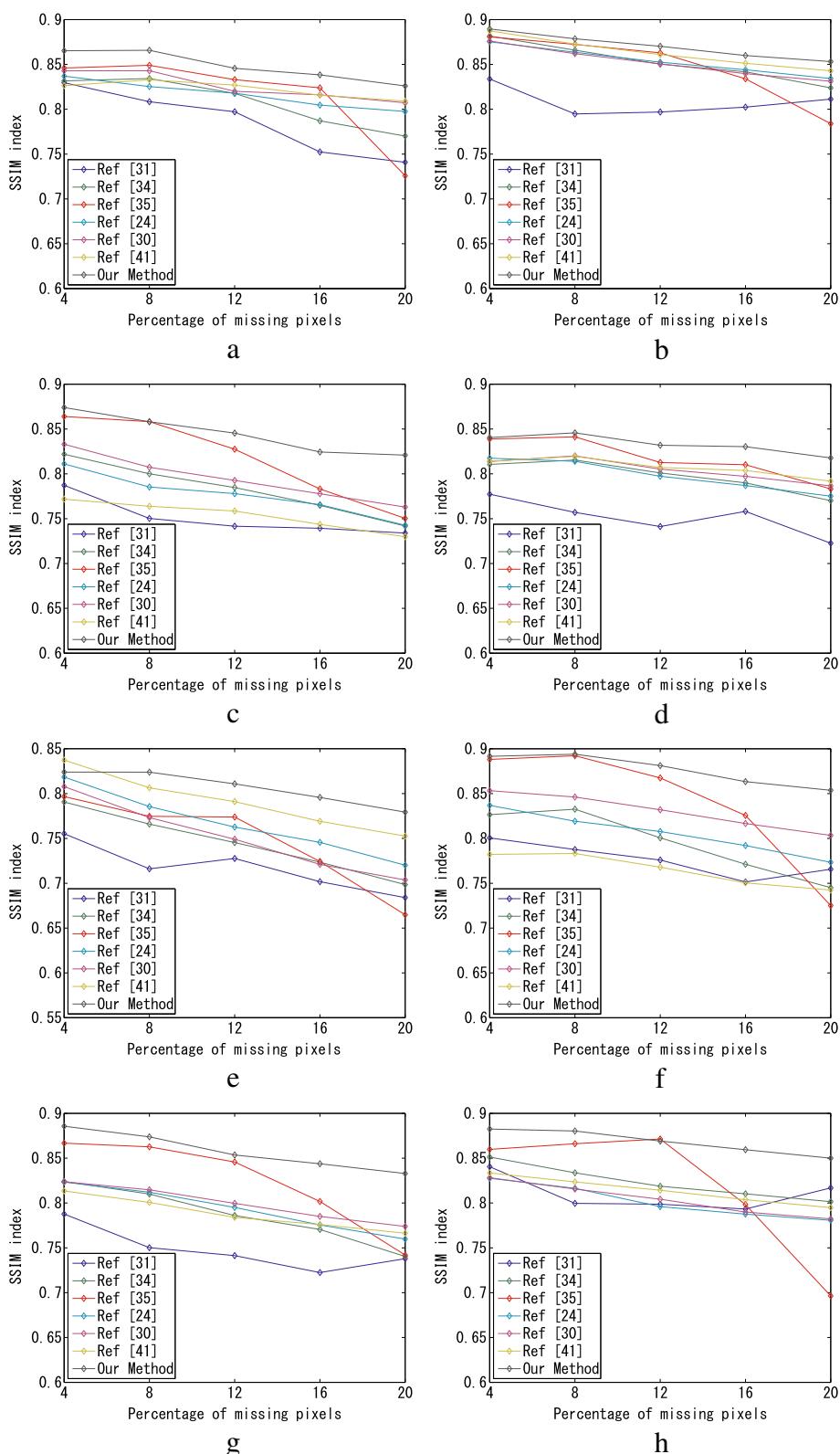
In recent years, several researchers of image quality assessment have also pointed out the problem that MSE and its variants cannot reflect some degradations [46,51]. Therefore, in order to tackle this problem, several criteria for determining image qualities have been proposed, the SSIM index being a representative criterion. In the proposed method, we focus on this criterion and realize inpainting that maximizes the SSIM index. Therefore, it is natural that the proposed method achieves the highest SSIM values. Note that even though the use of the SSIM index for inpainting is effective, it is difficult to perfectly determine the order of inpainting performance that is the same as the subjective evaluation. This means that ranking of inpainting performance that perfectly reflects subjective evaluation is difficult, and further improvement is necessary in future work.

As quantitative evaluation, we have shown PSNR and SSIM index of the results by our method and other existing methods. Next, we focus on the computation cost of the proposed method. We first compare the computation times of the proposed method and other multivariate analysis-based methods in [31] and [35]<sup>b</sup>. The average computation time for obtaining the results of images 1 to 11 by our method was about 342.1 s. Then the proposed method is about 0.78 to 3.1 times (1.6 times on average) slower than the method in [31]. Note that the ratio smaller than one means that the computation time of the proposed method is shorter. In the method in [31], the procedure for the inpainting is simple since it only needs the computation of the eigenvector matrix and the calculation of the back projection for lost pixels. Therefore, the

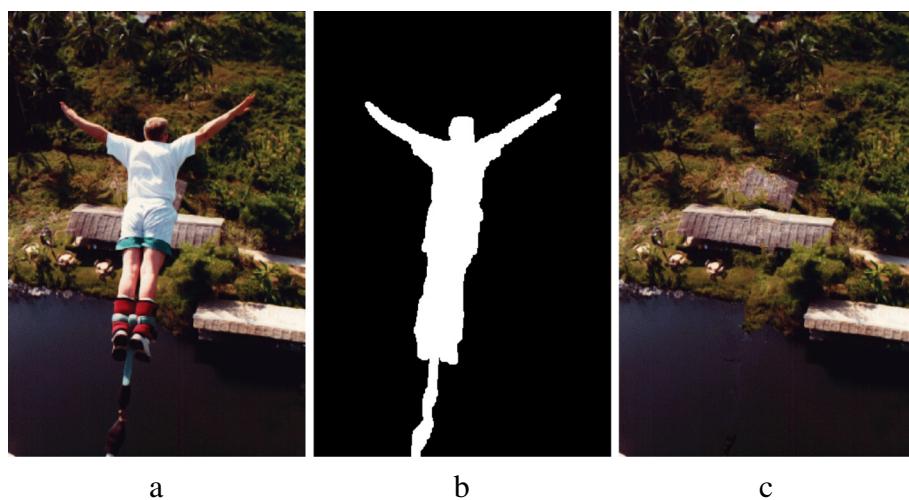
fast computation can be realized. On the other hand, the proposed method is about 1.2 to 4.7 times (2.9 times on average) faster than the method in [35]. In this method, the kernel PCA is adopted, and we have to calculate the projection onto the nonlinear subspace using the kernel trick, i.e., we cannot perform the direct projection. Furthermore, in this approach, the classification of the target patch including missing areas is performed, and thus, the inpainting procedures are performed for all clusters. Therefore, this needs high computation costs. Furthermore, the computation time of our method is about 4.3 to 12.5 times (7.4 times on average) longer than that of the exemplar-based method in [24]. Note that the method in [30] used as the comparative method in the experiments drastically improves the computation costs of [24], and it also introduces the GPU implementation. The CPU version improving the computation costs of [24] has also been proposed by the same authors [62]. In [62], Kwok et al. reported inpainting that was about 15 to 50 times faster than that of the method in [24].

In addition, compared with the MSE-based inpainting approach, which calculates the optimal sparse representation coefficients based on the MSE, the proposed method requires complex optimization procedures as shown in the previous section. In the MSE-based approach, it is well known that the normal equation can be simply solved, and it is much simpler than our method. It is therefore necessary to improve the speed of computation by introducing some alternative approaches into our inpainting method. This topic will be investigated in subsequent studies.

Note that in the above experiment, we used the difficult condition, i.e., larger size patches, in order to make the difference of the inpainting performance between our method and the existing methods clearer. Next, we show other different experimental results obtained using conditions which were adopted in each paper. This means that the conditions of the existing methods were determined



**Figure 12 Relationship between ratio of missing pixels and SSIM index of inpainting results obtained by each method. (a-h)** They respectively show the results obtained from images 4 to 11 shown in Figures 10 and 11.



**Figure 13** Inpainting example 1 obtained by applying the proposed method to larger missing areas. **(a)** Original image ( $342 \times 512$  pixels, 11.2% loss), **(b)** flag image of (a), and **(c)** results obtained by the proposed method.

according to their papers. In the new experiments, we used the eight test images shown in Figures 10 and 11 and randomly added missing blocks of size  $8 \times 8$  pixels with the changing ratio of the missing pixels. Figure 12 shows the relationship between the ratio of the missing pixels and the SSIM index calculated from the reconstructed image. From these results, we can see that the proposed method tends to output better results than those of the existing methods.

### 5.3 Inpainting of larger missing areas

Finally, in Figures 13, 14, 15, and 16, we show some examples obtained by performing inpainting for larger missing areas based on the proposed method. Note that in these experiments, we performed inpainting of larger missing areas by the proposed method including one simple additional procedure. The details are shown as follows. First, for a target patch selected on the basis of the patch priority shown in Equation 43, the proposed method performs the inpainting shown in Section 4.2 and obtains the result  $\hat{y}$ . Next, as an additional procedure, we search for the optimal known patch, which is best matched to the obtained result  $\hat{y}$ , using the SSIM index from the target image, and the selected known patch is used as the final output. Then, by performing the above procedures for all patches selected according to patch priority, the whole missing areas can be reconstructed.

The above scheme is similar to existing methods that simply select only the best matched examples, but the difference is shown below. In existing methods using only the best matched examples, the best matched patch is selected by monitoring errors in the known neighboring areas around the missing areas. On the other hand, the proposed method performs reconstruction of the patches

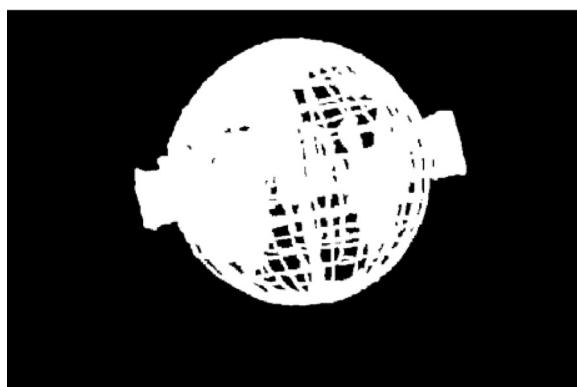
based on SSIM-based sparse representation, and then, the examples that are best matched to the reconstructed patches are selected using the SSIM index, i.e., the best matched examples are selected from well-approximated reconstruction patches. This is the biggest difference between the existing methods and the proposed method.

It should be noted that although the proposed method can perform accurate reconstruction of patches, the obtained results tend to include color that is not included within the target image. This is because the proposed method does not adopt any specific procedures to avoid spurious color. Therefore, in the experiments on reconstruction of larger missing areas, we adopted the above scheme to avoid the propagation of spurious color.

From the obtained results, we can confirm that the proposed method enables successful inpainting of such large missing areas. Note that the images shown in Figures 13, 14, 15, and 16 are used as test images in several papers such as in [23,24,30] and [41]. Furthermore, since the flag images that correspond to Figures 13b, 14b, 15b, and 16b are generated in each paper, i.e., positions of missing areas are different from each other in those papers, we show discussion by comparing the results obtained by our method shown in Figures 13, 14, 15, and 16 and the results shown in those papers. From the results shown in these figures, we can see that the proposed method achieves comparable performance or some improvements, though it should be noted that since we do not have ground truth images for these test images, we perform subjective evaluation. Specifically, as shown in Figure 15, the proposed method and the methods in [23] and [41] can achieve visually pleasant results. In this test image, since structural and textural components are simple and the percentage of missing areas is relatively small, it is easier to achieve



a



b



c

**Figure 14** Inpainting example 2 obtained by applying the proposed method to larger missing areas. (a) Original image (818 × 545 pixels, 24.8% loss), (b) flag image of (a), and (c) results obtained by the proposed method.



a



b

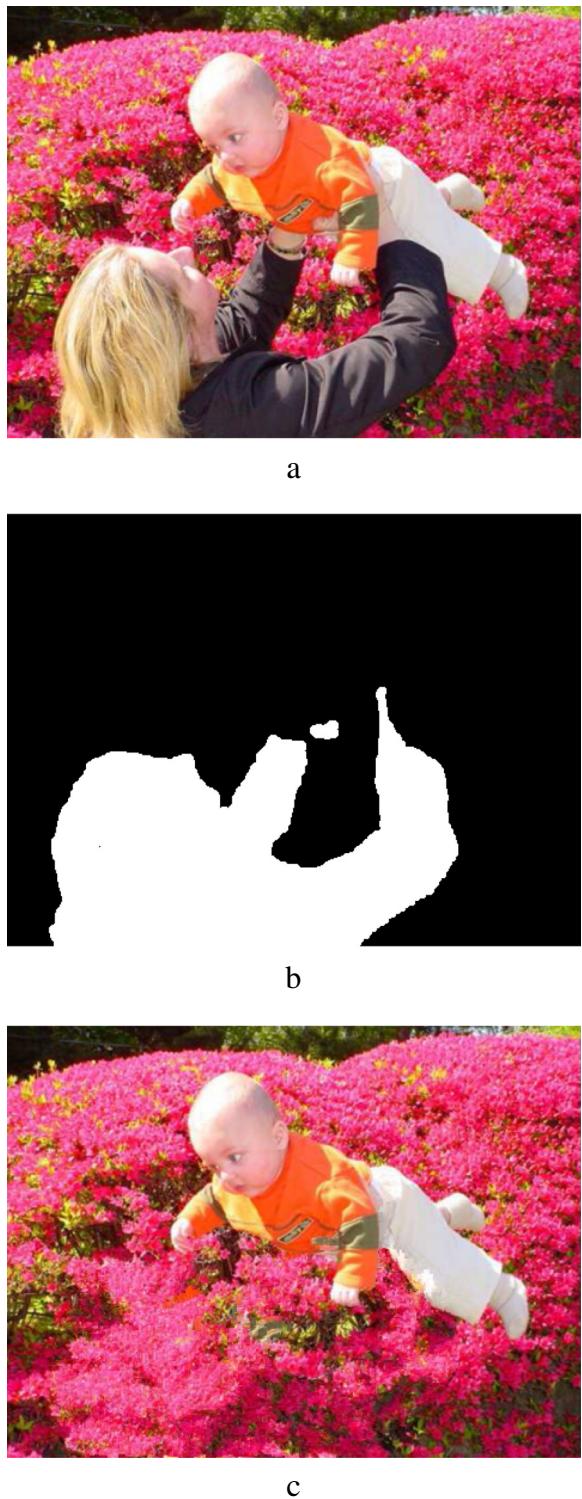


c

**Figure 15** Inpainting example 3 obtained by applying the proposed method to larger missing areas. (a) Original image (818 × 546 pixels, 11.5% loss), (b) flag image of (a), and (c) results obtained by the proposed method.

successful inpainting. Similarly, Figure 13 shows that successful inpainting could be achieved by our method and the methods in [24], [30], and [41], and improvement by our method can be confirmed in some areas. However, it should be noted that reconstruction of structural components, i.e., edges, by [41] can be realized more accurately.

The biggest difference between [41] and other works including our method is priority estimation. Thus, by introducing an improved priority estimation scheme, the performance of the proposed method will be improved. Furthermore, Figure 16 shows that results obtained by our

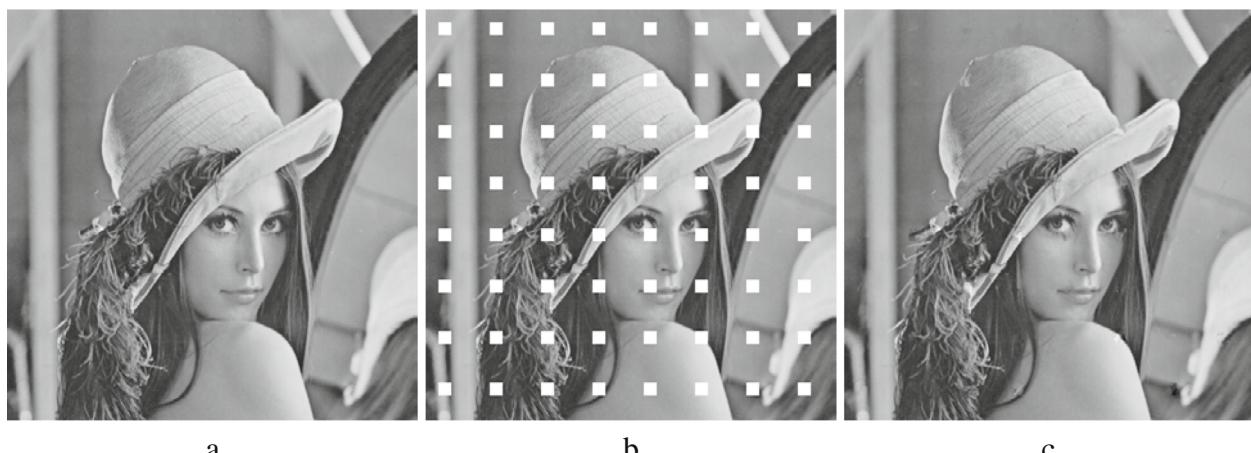


**Figure 16** Inpainting example 4 obtained by applying the proposed method to larger missing areas. **(a)** Original image ( $480 \times 362$  pixels, 25.2% loss), **(b)** flag image of **(a)**, and **(c)** results obtained by the proposed method.

method are comparable to results in [24] and [30]. Note that the flag images of this test image are different from each other in these methods, and we found that performance was affected by generation of flag images. This was also observed in the image shown in Figure 14.

As shown in the above discussion, it becomes difficult in the proposed method to perform successful structure reconstruction. In order to understand this problem easily, we show some examples in Figures 17 and 18. From the two images ( $512 \times 512$  pixels) shown in Figures 17a and 18a, we artificially added missing blocks ( $16 \times 16$  pixels) to obtain the corrupted images in Figures 17b and 18b. As shown in the results reconstructed by the proposed method in Figures 17c and 18c, it can be seen that texture regions and simple structure regions can be reconstructed successfully. On the other hand, in some complex structure regions including several directional edges simultaneously, it becomes difficult to perfectly recover those structure components by our method. This is because the proposed method considers the structure components only in the patch priority determination. This means the inpainting algorithm in the proposed method is optimized only for the texture reconstruction.

In order to simultaneously reconstruct the structure and texture regions, several methods have been proposed [23,25,42]. The method in [23] proposed a fragment-based algorithm which could preserve both structures and textures. A confidence map is used to determine which pixels have more surrounding information available. The reconstruction is performed from more confident pixels and is proceeded in a multiscale fashion from coarse to fine. Furthermore, a similar image fragment is found and copied to current unknown location, where a fragment is a circular neighborhood, and its radius is defined adaptive to its underlying structure. In contrast to the above advantage, it is reported in [24,41] that this algorithm is extremely slow and may introduce blurring artifacts. The fragment is selected based on the absolute distance, and this tends to cause the problem, i.e., the blurring artifacts, similar to that caused when using the MSE-based distance. The method in [42] introduced a sparse representation model representing both structure and texture components to realize their simultaneous reconstruction. On the other hand, this method is based on least-square approximation, and the problem of using the MSE may occur. Therefore, by introducing this simultaneous representation model into the proposed SSIM-based approach, successful reconstruction can be expected. Furthermore, the method in [25] introduced interactive image editing tools to realize highly accurate structure reconstruction. Since the guide for the reconstruction can be provided by users, this improves the inpainting performance. Although this approach does not realize the perfectly automatic image



**Figure 17** Inpainting results obtained by applying the proposed method to the test image 'Lena'. **(a)** Original image ( $512 \times 512$  pixels), **(b)** corrupted image including missing blocks ( $16 \times 16$  pixels), and **(c)** results obtained by the proposed method.

inpainting, it will also improve the performance of the proposed method by adopting the interactive image editing tools.

## 6 Conclusions

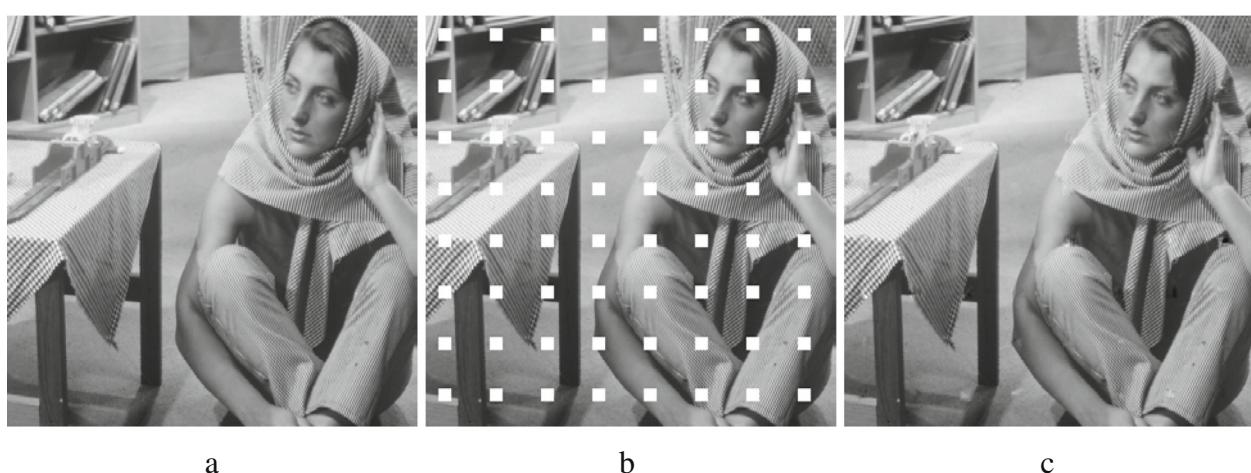
In this paper, we have presented an inpainting method based on sparse representations optimized with respect to a perceptual metric. Using sparse representation, the proposed method adaptively provides subspaces optimal for reconstructing target patches including missing areas. In this approach, the SSIM-based criterion is introduced into calculation of the dictionary and inpainting algorithm. This enables perceptually optimized inpainting, and successful results can be obtained by the proposed method.

Although the proposed method can reconstruct large missing regions without blurring artifacts, it has more computational complexity than other existing approaches and also generates some artifacts in the output image as shown in Figure 14. The computation cost and some artifacts caused by the proposed method should be concerned and solved in the future work.

Furthermore, extension of the algorithm to reconstruction of other types of missing image data is desirable for various applications. These topics will be future works and results will be presented in subsequent reports.

## Endnotes

<sup>a</sup>In this paper, signal-atoms are simply referred to as 'atoms' hereafter according to [38].



**Figure 18** Inpainting results obtained by applying the proposed method to the test image 'Barbara'. **(a)** Original image ( $512 \times 512$  pixels), **(b)** corrupted image including missing blocks ( $16 \times 16$  pixels), and **(c)** results obtained by the proposed method.

<sup>b</sup>The experiments were performed on a personal computer using Intel(R) Core(TM) i7 950 CPU 3.06 GHz with 8.0 GB RAM. The implementation was performed using MATLAB.

## Appendix

In this appendix, we show the details of the derivations in Equations 26 and 27. Since the optimization problem in Equation 25 is still nonconvex, it is converted into a quasi-convex optimization problem as follows:

$$\hat{\mathbf{x}}_{ij}^{(t)} \left( \rho_{ij}^{(t)} \right) = \arg \max_{\mathbf{x}_{ij}^{(t)} \in \mathbf{R}^t} \left( \frac{2\mathbf{k}_{ij}^{(t)'} \mathbf{x}_{ij}^{(t)} + C_2}{\sigma_{\mathbf{y}_i}^2 + \mathbf{x}_{ij}^{(t)'} \mathbf{K}_{ij}^{(t)} \mathbf{x}_{ij}^{(t)} + C_2} \right)$$

subject to  $\boldsymbol{\mu}_{\mathbf{D}_{ij}^{(t)}}' \mathbf{x}_{ij}^{(t)} = \rho_{ij}^{(t)}$ ,

$\Leftrightarrow$

$\min : \tau$

subject to  $\left[ \begin{array}{l} \max : \left( \frac{2\mathbf{k}_{ij}^{(t)'} \mathbf{x}_{ij}^{(t)} + C_2}{\sigma_{\mathbf{y}_i}^2 + \mathbf{x}_{ij}^{(t)'} \mathbf{K}_{ij}^{(t)} \mathbf{x}_{ij}^{(t)} + C_2} \right) \leq \tau \\ \text{subject to } \boldsymbol{\mu}_{\mathbf{D}_{ij}^{(t)}}' \mathbf{x}_{ij}^{(t)} = \rho_{ij}^{(t)} \end{array} \right]$ ,

$\Leftrightarrow$

$\min : \tau$

subject to

$\left[ \begin{array}{l} \min : \left[ \tau \left( \sigma_{\mathbf{y}_i}^2 + \mathbf{x}_{ij}^{(t)'} \mathbf{K}_{ij}^{(t)} \mathbf{x}_{ij}^{(t)} + C_2 \right) - \left( 2\mathbf{k}_{ij}^{(t)'} \mathbf{x}_{ij}^{(t)} + C_2 \right) \right] \geq 0 \\ \text{subject to } \boldsymbol{\mu}_{\mathbf{D}_{ij}^{(t)}}' \mathbf{x}_{ij}^{(t)} = \rho_{ij}^{(t)} \end{array} \right]. \quad (46)$

Since minimization of  $\tau$  is the same as finding the least upper bound of Equation 25, the first equivalence relationship holds. The second equivalence relationship holds since the denominator in Equation 25 is strictly positive, allowing us to multiply through and rearrange terms. In this way, we can derive Equation 26. Then,  $\tau$  becomes a true upper bound if

$$S_\tau = \tau \left( \sigma_{\mathbf{y}_i}^2 + \mathbf{x}_{ij}^{(t)'} \mathbf{K}_{ij}^{(t)} \mathbf{x}_{ij}^{(t)} + C_2 \right) - \left( 2\mathbf{k}_{ij}^{(t)'} \mathbf{x}_{ij}^{(t)} + C_2 \right) \quad (47)$$

optimized in Equation 46 has a nonnegative optimal value, and the optimal vector  $\hat{\mathbf{x}}_{ij}^{(t)} \left( \rho_{ij}^{(t)} \right)$  in Equation 25 can be obtained. Thus, by applying the Lagrange multiplier approach to the above equation under the constraint  $\boldsymbol{\mu}_{\mathbf{D}_{ij}^{(t)}}' \mathbf{x}_{ij}^{(t)} = \rho_{ij}^{(t)}$ , Equation 27 can be obtained.

### Competing interests

The authors declare that they have no competing interests.

### Acknowledgements

This work was partly supported by Grant-in-Aid for Scientific Research (B) 25280036 and Grant-in-Aid for Young Scientists (B) 22700088, Japan Society for the Promotion of Science (JSPS).

Received: 14 June 2013 Accepted: 15 November 2013  
 Published: 5 December 2013

### References

1. R Yan, L Shao, Y Liu, Nonlocal hierarchical dictionary learning using wavelets for image denoising. *IEEE Trans. Image Process.* **22**(12), 4689–4698 (2013)
2. L Shao, R Yan, X Li, Y Liu, From heuristic optimization to dictionary learning: A review and comprehensive comparison of image denoising algorithms. *IEEE Trans. Cybern.* (2013). doi:10.1109/TCYB.2013.2278548
3. A Buades, B Coll, J Morel, A review of image denoising algorithms, with a new one. *Multiscale Model. Simul.* **4**(2), 490–530 (2005)
4. L Shao, H Zhang, G de Haan, An overview and performance evaluation of classification-based least squares trained filters. *IEEE Trans. Image Process.* **17**(10), 1772–1782 (2008)
5. PC Hansen, JG Nagy, DP O’Leary, *Deblurring Images: Matrices, Spectra, and Filtering (Fundamentals of Algorithms)*. (Society for Industrial and Applied Mathematics, Philadelphia, 2006)
6. Z Tauber, ZN Li, M Drew, Review and preview: disocclusion by inpainting for image-based rendering. *IEEE Trans. Syst., Man, Cybern., Part C: Appl. Rev.* **37**(4), 527–540 (2007)
7. S Masnou, J Morel, Level lines based disocclusion. *Proc. IEEE Int. Conf. Image Process. (ICIP)* **3**, 259–263 (1998)
8. M Bertalmio, G Sapiro, V Caselles, C Ballester, Image inpainting, in *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques, (SIGGRAPH’00)* (ACM New York, 2000), pp. 417–424
9. M Bertalmio, A Bertozzi, G Sapiro, Navier-stokes, fluid dynamics, and image and video inpainting. *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.* **1**, 355–362 (2001)
10. TF Chan, J Shen, Nontexture inpainting by curvature-driven diffusions. *J. Vis. Commun. Image Representation* **12**(4), 436–449 (2001)
11. T Ogawa, M Haseyama, H Kitajima, Restoration method of missing areas in still images using GMRF model. *IEEE Int. Symp. Circuits Syst. (ISCAS)* **5**, 4931–4934 (2005)
12. A Rares, MJT Reinders, J Biemond, Edge-based image restoration. *IEEE Trans. Image Process.* **14**(10), 1454–1468 (2005)
13. M-F Auclair-Fortier, D Ziou, A global approach for solving evolutive heat transfer for image denoising and inpainting. *IEEE Trans. Image Process.* **15**(9), 2558–2574 (2006)
14. M Bertalmio, Strong-continuation, contrast-invariant inpainting with a third-order optimal PDE. *IEEE Trans. Image Process.* **15**(7), 1934–1938 (2006)
15. J Liu, M Li, F He, A novel inpainting model for partial differential equation based on curvature function. *J. Multimedia* **7**(3), 239–246 (2012)
16. M Wang, B Yan, KN Ngan, An efficient framework for image/video inpainting. *Signal Process: Image Commun.* **28**(7), 753–762 (2013)
17. F Qi, J Han, P Wang, G Shi, F Li, Structure guided fusion for depth map inpainting. *Pattern Recognit. Lett.* **34**, 70–76 (2013)
18. C C Ballester, M Bertalmio, V Caselles, G Sapiro, Filling-in by joint interpolation of vector fields and gray levels. *IEEE Trans. Image Process.* **10**(8), 1200–1211 (2001)
19. A Kokaram, A statistical framework for picture reconstruction using 2D AR models. *Image Vis. Comput.* **22**(2, 1), 165–171 (2004)
20. M Bertalmio, L Vese, G Sapiro, S Osher, Simultaneous structure and texture image inpainting. *IEEE Trans. Image Process.* **12**(8), 882–889 (2003)
21. AA Efros, TK Leung, Texture synthesis by nonparametric sampling. *Proceedings of the Seventh IEEE International Conference on Computer Vision* **2**, 1033–1038 (1999)
22. LW Wey, M Levoy, Fast texture synthesis using tree-structured vector quantization, in *Proceedings of SIGGRAPH 2000* (ACM New York, 2000), pp. 479–488
23. I Drori, D Cohen-Or, H Teshurun, Fragment-based image completion, in *Proceedings of SIGGRAPH 2003* (ACM New York, 2003), pp. 303–312
24. A Criminisi, P Perez, K Toyama, Region filling and object removal by exemplar-based image inpainting. *IEEE Trans. Image Process.* **13**(9), 1200–1212 (2004)
25. C Barnes, E Shechtman, A Finkelstein, DB Goldman, PatchMatch: a randomized correspondence algorithm for structural image editing. *ACM Trans. Graph. (Proc. SIGGRAPH)* (2009). doi:10.1145/1531326.1531330
26. Q Zhang, J Lin, Exemplar-based image inpainting using color distribution analysis. *J. Inf. Sci. Eng.* **28**(4), 641–654 (2012)
27. T Shibata, A Iketani, S Senda, Fast and structure-preserving image inpainting based on probabilistic structure estimation. *IEICE Trans. Inf. Syst.* **95**(7), 1731–1739 (2012)

28. O Le Meur, C Guillemot, ed. by A Fitzgibbon, S Lazebnik, P Perona, Y Sato, and C Schmid. Super-resolution-based inpainting, in *Computer Vision - ECCV 2012. 12th European Conference on Computer Vision, Florence, 7-13 October 2012, Proceedings, Part VI* (Springer, Heidelberg, 2012), pp. 554–567
29. O Le Meur, M Ebdelli, C Guillemot, Hierarchical super-resolution-based inpainting. *Image Process. IEEE Trans.* **22**(10), 3779–3790 (2013)
30. TH Kwok, H Sheung, CCL Wang, Fast query for exemplar-based image completion. *IEEE Trans. Image Process.* **19**(12), 3106–3115 (2010)
31. T Amano, Y Sato, Image interpolation using BPLP method on the eigenspace. *Syst. Comput. Japan* **38**, 87–96 (2007)
32. B Schölkopf, S Mika, CJC Burges, P Knirsch, KR Müller, G Rätsch, AJ Smola, Input space versus feature space in kernel-based methods. *IEEE Trans. Neural Netw.* **10**(5), 1000–1017 (1999)
33. S Mika, B Schölkopf, A Smola, KR Müller, M Scholz, G Rätsch, Kernel PCA and de-noising in feature spaces. *Adv. Neural Inf. Process. Syst.* **11**, 536–542 (1999)
34. KI Kim, MO Franz, B Schölkopf, Iterative kernel principal component analysis for image modeling. *IEEE Trans. Pattern Anal. Mach. Intell.* **27**(9), 1351–1366 (2005)
35. T Ogawa, M Haseyama, Missing intensity interpolation using a kernel PCA-Based POCS algorithm and its applications. *IEEE Trans. Image Process.* **20**(2), 417–432 (2011)
36. M Aharon, M Elad, A Bruckstein, K-SVD: an algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Trans. Signal Process.* **54**(11), 4311–4322 (2006)
37. M Elad, M Aharon, Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Trans. Image Process.* **15**(12), 3736–3745 (2006)
38. J Mairal, M Elad, G Sapiro, Sparse representation for color image restoration. *IEEE Trans. Image Process.* **17**, 53–69 (2008)
39. B Wohlberg, Inpainting with sparse linear combinations of exemplars, in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2009)* (IEEE Piscataway, 2009), pp. 689–692
40. B Shen, W Hu, Y Zhang, YJ Zhang, Image inpainting via sparse representation, in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2009)* (IEEE Piscataway, 2009), pp. 697–700
41. Z Xu, J Sun, Image inpainting by patch propagation using patch sparsity. *IEEE Trans. Image Process.* **19**(5), 1153–1165 (2010)
42. M Elad, JL Starck, P Querre, D Donoho, Simultaneous cartoon and texture image inpainting using morphological component analysis (MCA). *Appl. Comput. Harmonic Anal.* **19**(3), 340–358 (2005)
43. M Turkan, C Guillemot, Locally linear embedding based texture synthesis for image prediction and error concealment, in *19th IEEE International Conference on Image Processing (ICIP)* (IEEE Piscataway, 2012), pp. 3009–3012
44. C Guillemot, M Turkan, OL Meur, M Ebdelli, Object removal and loss concealment using neighbor embedding methods. *Signal Process.: Image Commun.* **28**(10), 1405–1419
45. T Takahashi, K Konishi, T Furukawa, Structured matrix rank minimization approach to image inpainting, in *IEEE 55th International Midwest Symposium on Circuits and Systems (MWSCAS)* (IEEE Piscataway, 2012), pp. 860–863
46. B Girod, ed. by AB Watson. What's wrong with mean-squared error?, in *Digital Images and Human Vision* (MIT Press Cambridge, 1993), pp. 207–220
47. Z Wang, AC Bovik, *Modern Image Quality Assessment*. (Morgan & Claypool Publishers, San Rafael, 2006)
48. N Damera-Venkata, TD Kite, WS Geisler, BL Evans, AC Bovik, Image quality assessment based on a degradation model. *IEEE Trans. Image Process.* **4**(4), 636–650 (2000)
49. HR Sheikh, AC Bovik, G de Veciana, An information fidelity criterion for image quality assessment using natural scene statistics. *IEEE Trans. Image Process.* **14**(12), 2117–2128 (2005)
50. HR Sheikh, AC Bovik, Image information and visual quality. *IEEE Trans. Image Process.* **15**(2), 430–444 (2006)
51. HR Sheikh, MF Sabir, AC Bovik, A statistical evaluation of recent full reference image quality assessment algorithms. *IEEE Trans. Image Process.* **15**(11), 3440–3451 (2006)
52. Z Wang, AC Bovik, HR Sheikh, EP Simoncelli, Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Process.* **13**(4), 600–612 (2004)
53. SS Channappayya, AC Bovik, C Caramanis, RW Heath, Design of linear equalizers optimized for the structural similarity index. *IEEE Trans. Image Process.* **17**(6), 857–872 (2008)
54. A Rehman, M Rostami, Z Wang, D Brunet, ER Vrscay, SSIM-inspired image restoration using sparse representation. *EURASIP J. Adv. Signal Process.* **2012**, 16 (2012)
55. G Davis, S Mallat, M Avellaneda, Adaptive greedy approximations. *J. Construct. Approx.* **13**, 57–98 (1997)
56. S Malitz, Z Zhang, Matching pursuits with time-frequency dictionaries. *IEEE Trans. Signal Process.* **41**(12), 3397–3415 (1993)
57. S Chen, SA Billings, W Luo, Orthogonal least squares methods and their applications to non-linear system identification. *Int. J. Contr.* **50**(5), 1873–1896 (1989)
58. YC Pati, R Rezaifar, PS Krishnaprasad, Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition. Conf. Rec. 27th Asilomar Conf. Signals, Syst. Comput. **1**, 40–44 (1993)
59. JA Tropp, Greed is good: algorithmic results for sparse approximation. *IEEE Trans. Inf. Theory* **50**(10), 2231–2242 (2004)
60. SS Chen, DL Donoho, MA Saunders, Automatic decomposition by basis pursuit. *SIAM Rev.* **43**, 129–159 (2001)
61. IF Gorodnitsky, BD Rao, Sparse signal reconstruction from limited data using FOCUSS: a re-weighted norm minimization algorithm. *IEEE Trans. Signal Process.* **45**(3), 600–616 (1997)
62. TH Kwok, CCL Wang, Interactive image inpainting using DCT based exemplar matching. *Adv. Vis. Comput., Lecture Notes Comput. Sci.* **5876/2009**, 709–718 (2009)

doi:10.1186/1687-6180-2013-179

**Cite this article as:** Ogawa and Haseyama: Image inpainting based on sparse representations with a perceptual metric. *EURASIP Journal on Advances in Signal Processing* 2013 **2013**:179.

## Submit your manuscript to a SpringerOpen® journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Immediate publication on acceptance
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► [springeropen.com](http://springeropen.com)