Multiple Subspace Model and Image-Inpainting Algorithm Based on Multiple Matrix Rank Minimization

Tomohiro TAKAHASHI†*, Katsumi KONISHI††, Kazunori URUMA†††, and Toshihiro FURUKAWA††††, Members

SUMMARY This paper proposes an image inpainting algorithm based on multiple linear models and matrix rank minimization. Several inpainting algorithms have been previously proposed based on the assumption that an image can be modeled using autoregressive (AR) models. However, these algorithms perform poorly when applied to natural photographs because they assume that an image is modeled by a position-invariant linear model with a fixed model order. In order to improve inpainting quality, this work introduces a multiple AR model and proposes an image inpainting algorithm based on multiple matrix rank minimization with sparse regularization. In doing so, a practical algorithm is provided based on the iterative partial matrix shrinkage algorithm, with numerical examples showing the effectiveness of the proposed algorithm.

key words: matrix rank minimization, image inpainting, multiple linear models

1. Introduction

This paper deals with the problem of image inpainting, which is a technique for restoring damaged regions of an image. Accordingly, we consider repairing an image with a large number of small dots missing or narrow lines missing. Indeed, image inpainting can be applied to a number of different fields, such as removing character strings overwritten on photographs, and improving the image quality of scanning sensors used in satellite remote sensing [1].

There are two approaches for detecting the missing pixels: the user provides the missing pixel information with the image, or the missing area is simultaneously detected during inpainting. In this paper, we assume that the missing region is provided as prior information. Various algorithms have been proposed for image inpainting such as exemplar based approaches [2]–[5], back projection approaches [6], [7], partial differential equation based approaches [8], [9] and deep learning based approaches [10], [11]. Exemplar based approaches recover pixels that are missing from an image (hereinafter referred to as the observed image) using either the observed image itself or a database of known images, with restoration achieved by finding a similar patch of pixels in an undamaged part of one of these images. In general, exemplar based approaches are effective at recovering missing pixels well even if the missing region is large. Back projection approaches assume that the complete image is represented by a linear combination of a few bases given by principal component analysis (PCA), and the missing pixels are restored using the coefficients of this linear combination. Ogawa et al. proposed an adaptive subspace approach that divides the image patch into multiple clusters and performs a nonlinear inverse projection for low dimensional space using kernel PCA for each cluster [7]. This approach can achieve high quality restoration if the patches in the observed image are completely undamaged and if, moreover, the clusters are adequately studied. In this paper, however, we assume that the observed image is missing more than half of its pixels, meaning that the problem setting of this paper differs from that of [7]. Deep learning based approaches have evolved in recent years [10], [11] and have achieved remarkable inpainting results, such as restoring a partly hidden face. However, the purpose of these algorithms is not necessarily to restore ground truth. Rather, the image is restored similar to the ground truth when inpainting techniques are used for the engineering applications such as satellite image processing [1]. Recently, several matrix rank minimization approaches have been proposed to achieve image inpainting. Szaier et al. proposed a Hankel operator approach to texture inpainting [12]. Moreover, structured matrix rank minimization based image inpainting algorithms have been proposed by [13], [14]. The authors proposed an inpainting algorithm based on an autoregressive (AR) model that estimates the values of the missing pixels and the model order simultaneously [13]. However, the restoration quality of the aforementioned inpainting algorithms is insufficient with respect to recovering general images because they assume that images are modeled using position-invariant linear systems with a fixed model order. Although an image inpainting algorithm using block-based model estimation has been proposed by [15], natural images are far too complicated to be represented with rectangular block moreover, in these algorithms, inpainting quality with the block algorithm depends on the given block size.

In order to improve inpainting quality, this paper introduces a multiple AR model and proposes a new multiple matrix rank minimization approach with sparse regulariza-
tion. Whereas a single AR model represents only one type of object texture, a multiple AR model can represent different textures for each object and it is, therefore, suitable for standard photographs. The Hankel matrices of an AR sequence span low dimensional subspace which differs from model to model. Therefore, the problem of identifying multiple models is equivalent to that of identifying multiple subspaces, which is also known as generalized principal component analysis (GPCA) [16], [17]. Because the estimation results of missing pixels and subspaces affects other estimations, image inpainting problem requires that identify multiple subspaces and restore missing pixels simultaneously.

The authors have already proposed an algorithm for simultaneously obtaining missing signals and subspaces using GPCA [18], [19]. However, this algorithm caused subspace overlapping and excessive switching, which is obviously inappropriate for general image restoration. This paper describes the image restoration algorithm based on GPCA [18] in more detail, proposing an algorithm to prevent overlaps and excessive switching using regularizations in the process. Numerical examples show the effectiveness of the proposed algorithm.

This paper is organized as follows; Sect. 2 presents the block Hankel matrix rank minimization based image inpainting algorithm and its limitations; Sect. 3, we propose the multiple matrix rank minimization algorithm based on subspace clustering with variation regularization. Numerical examples are shown in Sect. 4, with conclusions given in Sect. 5.

2. Matrix Rank Minimization Approach

This section explains the image inpainting algorithm proposed by [13].

Let \( I_{i,j} \in [0, 1] \) denote the intensity of the \((i, j)\)th pixel in the \(M \times N\) image \(I\). In the approach based on the AR model, we assume that the image is modeled by an AR model as follows,

\[
\sum_{m=-K}^{K} \sum_{l=-K}^{K} a_{l,m} I_{i+l,j+m} = 0, \tag{1}
\]

where \(K\) and \(a_{l,m}\) denote the model order and the \((l,m)\)th model coefficient, respectively. Accordingly, the AR model based approach estimates the model coefficients and the values of the missing pixels simultaneously using a given model order \(K\). In other words, the restoration quality of this approach strongly depends on \(K\). To overcome this dependency, a structured matrix rank minimization based algorithm is proposed [13], which formulates the image inpainting problem as follows,

\[
\text{Minimize} \quad \text{rank} X \\
\text{subject to} \quad X \in \mathcal{H} \cap I, \tag{2}
\]

where \(I\) denotes the matrices set satisfying \(x_{i,j} \in [0, 1]\) and \(x_{i,j} = I_{i,j}\) for \((i, j) \in \Omega\) using the index set of uncorrupted pixels \(\Omega\). In the above problem, \(\mathcal{H}\) denotes the set of Hankel-like structured matrices \(X\) defined by

\[
X = \begin{bmatrix}
x_{K+1,K+1} & \cdots & x_{K+1,N-K} \\
\vdots & \ddots & \vdots \\
x_{K+2,K+1} & \cdots & x_{K+2,N-K} \\
\vdots & \ddots & \vdots \\
x_{M-K,N-K} & \cdots & x_{M-K,N-K}
\end{bmatrix} \in \mathcal{H} \subset \mathbb{R}^{(M-2\hat{K})(N-2\hat{K}) \times (2\hat{K}+1)^2},
\]

where \(x_{i,j}\) is a row vector generated by the vectorizing part of the \((2\hat{K}+1) \times (2\hat{K}+1)\) image centered on \((i, j)\) and defined by

\[
x_{i,j} = \begin{bmatrix} x_{i-K,j-K} & x_{i-K,j-K+1} & \cdots & x_{i-K,j+K} \\
x_{i-K+1,j-K} & x_{i-K+1,j-K+1} & \cdots & x_{i-K+1,j+K} \\
\vdots & \vdots & \ddots & \vdots \\
x_{i+K,j-K} & x_{i+K,j-K+1} & \cdots & x_{i+K,j+K}
\end{bmatrix} \in \mathbb{R}^{(2\hat{K}+1)^2},
\]

and \(\hat{K}\) is a given upper bound of the model order. If \(K = \hat{K}\), (1) is equal to,

\[
Xa = 0, \quad a = [a_{l,m}] \in \mathbb{R}^{(2\hat{K}+1)^2}. \tag{5}
\]

Because the rank of \(X\) is an increasing function with respect to the model order, we can estimate a valid model order by finding the missing elements of \(X\) such that its rank is minimized; that is, problem (2) estimates missing pixels by minimizing \(\text{rank} X\) without estimating the model coefficients \(a_{l,m}\). This problem is difficult to solve because the matrix rank minimization problems are NP hard in general, we consider the relaxed problem of rank minimization problem. Although nuclear norm minimization is usually used to address the relaxation problem, this paper uses truncated nuclear norm minimization instead since it has a better performance [20], [21] despite the fact that it is not a convex function. The truncated nuclear norm \(\|X\|_{s,r}\) is the sum of nondominant singular values, which can be defined as follows,

\[
\|X\|_{s,r} = \sum_{i=r+1}^{\text{end}} \sigma_i \tag{6}
\]

where \(\sigma_i\) denotes the \(i\)th greatest singular value of matrix \(X\). Accordingly, the inpainting problem (2) can be expressed as follows,

\[
\text{Minimize} \quad \|X\|_{s,r} \\
\text{subject to} \quad X \in \mathcal{H} \cap I, \tag{7}
\]

Because the rank minimization based image inpainting algorithm assumes the position invariant model (1), its
restoration quality is insufficient to inpaint actual photographs. Some inpainting algorithms divide entire images into smaller blocks and apply image inpainting algorithms to each block, assuming that each block can be modeled by a single AR model. However, natural images are too complicated to be represented by rectangular blocks and the inpainting quality is dependent on given block size. To improve the inpainting quality, this work assumes that an image is modeled by multiple AR models.

3. Multiple Matrix Rank Minimization Approach

3.1 Subspace Clustering Approach

To improve inpainting quality, this paper introduces a multiple AR model and formulates the inpainting problem as a multiple matrix rank minimization problem. Figure 1 shows an overview of the proposed approach, wherein an observed image is divided into various regions, that are recovered independently. Because the results of image clustering and the recovered image are both uncertain at first, we update them alternately.

First, focusing on image clustering, we assume that an entire image is given without any missing pixels. The image clustering problem is formulated as the following multiple subspace identification problem proposed in [17],

\begin{align}
\text{Find} & \quad D_1, D_2, \ldots, D_L, a_1, a_2, \ldots, a_L \\
\text{subject to} & \quad \|D_iXa_i\|_2^2 \leq \epsilon, i = 1, 2, \ldots, L, \\
& \quad \sum_{i=1}^L D_i = E, \\
& \quad D_i \in \mathcal{D}, i = 1, 2, \ldots, L, \\
& \quad (D_i)_{j,j} = \{0, 1\}, \quad j = 1, 2, \ldots, L.
\end{align}

(8)

where $E, a_i, \mathcal{D}, (\cdot)_{j,j}, \epsilon$ and $L$ denote the identity matrix, the coefficient vector of the $i$th AR model, a set of diagonal matrices, the $(i, j)$th element of the matrix, a positive small constant, and the number of AR models, respectively. This approach divides an entire image into $L$ AR models using $D_i$ as the model selector matrix for the $i$th AR model, that is, the $i$th row vector of $X$ belongs to the $i$th AR model if $(D_i)_{j,j} \neq 0$.

Because the model order of each AR model is unknown, this paper takes a rank minimization approach and its truncated nuclear norm relaxation approach, wherein the image clustering problem is formulated as follows,

\begin{align}
\text{Find} & \quad D_1, D_2, \ldots, D_L \\
\text{subject to} & \quad \|D_iXa_i\|_{r_i} \leq \epsilon, i = 1, 2, \ldots, L, \\
& \quad \sum_{i=1}^L D_i = E, \\
& \quad D_i \in \mathcal{D}, i = 1, 2, \ldots, L, \\
& \quad (D_i)_{j,j} = \{0, 1\}.
\end{align}

(9)

Next we consider the problem of clustering and inpainting simultaneously. To solve this problem, this paper introduces the following iterative multiple subspace based signal recovery algorithm [18], [19].

\begin{align}
\text{Minimize} & \quad \sum_{i=1}^L \|D_iXa_i\|_{r_i} \\
\text{subject to} & \quad \sum_{i=1}^L D_i = E, \\
& \quad (D_i)_{j,j} = \{0, 1\}, \\
& \quad D_i \in \mathcal{D}, i = 1, 2, \ldots, L, \\
& \quad X \in \mathcal{H} \cap \mathcal{I}.
\end{align}

(10)

This problem is difficult to solve because $(D_i)_{j,j}$ has a 0-1 constraint. Therefore we consider the following relaxation problem,

\begin{align}
\text{Minimize} & \quad \sum_{i=1}^L \|D_iXa_i\|_{r_i} \\
\text{subject to} & \quad \sum_{i=1}^L D_i = E, \\
& \quad D_i \in \mathcal{D}, i = 1, 2, \ldots, L, \\
& \quad X \in \mathcal{H} \cap \mathcal{I}.
\end{align}

(11)

Because this problem does not consider the condition $(D_i)_{j,j} \in \{0, 1\}$, the obtained $(D_i)_{j,j}$ in $D_i$ usually does not equal zero or one; rather, it will have different values even if the corresponding pixels belong to the same object. Figure 2 shows a visualization of the model selectors $(D_i)_{j,j}$ obtained by solving (11). The values of $(D_i)_{j,j}$ are visualized as gray scale images, where black and white pixels denote zero and one, respectively. We can see many gray pixels, that is, lots of $(D_i)_{j,j}$ are not nearly equal to 0 or 1. To obtain $(D_i)_{j,j}$ values approximal to zero or one, this paper introduces two kinds of regularization.
3.2 Sparse Optimization Approach for Model Selection

As mentioned in the previous subsection, the estimation algorithm of the model selector $D_i$ proposed in [18] is inadequate. Moreover, we know that each element of $D_i$ must be close to zero or one so that one AR model represents one texture type, however, $D_i$ obtained from (11) causes excessive model switching, that is, an area represented by a single texture cannot be represented by an AR model. To overcome these problems, this paper introduces the sparse regularizations and formulates the image inpainting problem as follows,

$$
\text{Minimize} \quad f(X, \{D_i\}_{i=1}^L)
$$

subject to

$$
\sum_{i=1}^L D_i = E,
$$

$$
D_i \in \mathcal{D}, i = 1, 2, \ldots, L,
$$

$$
X \in \mathcal{H} \cap I,
$$

where

$$
f(X, \{D_i\}_{i=1}^L) = \sum_{i=1}^L \|D_iX\|_{\ast, r_1} + \gamma \sum_{i=1}^L \|\text{diag}(D_i)\|_1 + \tau \sum_{i=1}^L \|V\text{diag}(D_i)\|_2^2.
$$

(13)

diag(\cdot) : R^{M \times M} \rightarrow R^M$, $\| \cdot \|_1$ and $V$ denote an operator to extract diagonal components, the $\ell_1$ norm of a vector, and a matrix to calculate the differences between neighboring values of entries in model selectors, respectively. $\gamma$ and $\tau$ are the given positive values, and $V$ is defined using the vertical and horizontal difference operator matrices $V_v$ and $V_h$ as follows,

$$
\begin{align*}
V_v &= \text{diag}(V_r, \ldots, V_s) \in R^{M(N-1) \times MN}, \\
V_h &= \{1 : i = j, -1 : i + M = j, 0 : \text{otherwise}\} \in R^{(M-1) \times MN},
\end{align*}
$$

(14)

The second term of $f$ denotes sparsity of $D_i$, and this term forces the number of $(D_i)_{jj}$ satisfying to be in $\{0, 1\}$ to be increased by increasing the sparsity of $D_i$. The third term denotes their variations and is derived from the assumption that neighboring pixels belong to the same AR model in general images. The number of models we can estimate is limited, and the pixels that can be represented by one model are mostly adjacent. The third term lets the neighboring pixels have the same value of $(D_i)_{jj}$ by decreasing the variety of $D_i$.

The problem (12) is nonconvex and difficult to solve because of the product of $D_i$ and $X$. To solve (12), we relax this problem as follows,

$$
\text{Minimize} \quad f_{\gamma}(X, \{Z_i\}_{i=1}^L, \{D_i\}_{i=1}^L)
$$

subject to

$$
X \in \mathcal{H} \cap I, \quad D_i \in \mathcal{D}, \sum_{i=1}^L D_i = E,
$$

where

$$
f_{\gamma}(X, \{Z_i\}_{i=1}^L, \{D_i\}_{i=1}^L) = \sum_{i=1}^L \|Z_i - D_iX\|_F^2 + \lambda \|Z_i\|_{\ast, r_1} + \gamma \|\text{diag}(D_i)\|_1 + \tau \frac{1}{2} \|V\text{diag}(D_i)\|_2^2.
$$

(16)

The objective function of the relaxed problem (15) can be decreased by the following three steps,

Step 1.

$$
\{Z_i^{(t+1)}\}_{i=1}^L \leftarrow \text{argmin}_{Z_i \in \mathcal{D}} f_{\gamma}(X^{(t)}, \{Z_i^{(t)}\}_{i=1}^L, \{D_i^{(t)}\}_{i=1}^L)
$$

Step 2.

$$
\{D_i^{(t+1)}\}_{i=1}^L \leftarrow \text{argmin}_{D_i \in \mathcal{D}} f_{\gamma}(X^{(t)}, \{Z_i^{(t+1)}\}_{i=1}^L, \{D_i\}_{i=1}^L)
$$

subject to

$$
D_i \in \mathcal{D}, \sum_{i=1}^L D_i = E
$$

Step 3.

$$
X^{(t+1)} \leftarrow \text{argmin}_{X} f_{\gamma}(X, \{Z_i^{(t+1)}\}_{i=1}^L, \{D_i^{(t+1)}\}_{i=1}^L)
$$

subject to

$$
X \in \mathcal{H} \cap I
$$

Trivially the above scheme decreases the objective function of (15). In Step 1, $Z_i^{(t+1)}$ can be obtained as the partial singular value soft thresholding operator $\mathcal{T}_{\mathcal{D}, \lambda}(D_i X^{(t)})$, which replaces the $j$th singular value soft thresholding operator $\mathcal{T}_\lambda(D_i X^{(t)})$ with max($\sigma_{ij} - \lambda$, 0) for $j \geq r_1 + 1$. This is because, from Lemma 1 of [21], it hold that $\mathcal{T}_\lambda(X) = \text{argmin} \lambda \|Z\|_{\ast, r_1} + \frac{1}{2} \|Z - X\|_F^2$.

We can easily confirm that

$$
\begin{align*}
f(X^{(t+1)}, \{Z_i^{(t+1)}\}_{i=1}^L, \{D_i^{(t+1)}\}_{i=1}^L) \\ \leq f(X^{(t)}, \{Z_i^{(t)}\}_{i=1}^L, \{D_i^{(t)}\}_{i=1}^L).
\end{align*}
$$

(17)
This scheme decreases the objective function of (15). In this paper, singular values greater than \( \alpha \) times the greatest singular value are defined as dominant singular values for \( \alpha < 1 \). \( r_i \) and \( \lambda \) can be obtained as follows,

\[
\begin{align*}
    r_i &= \text{argmin}_{j} \sigma_{i,j} \text{ s.t. } \sigma_{i,j} \geq \alpha \sigma_{i,1} \\
    \lambda &= \lambda_0 \sigma_{i,j},
\end{align*}
\]

where \( \lambda_0 \) denotes a positive constant. The problem used to obtain \( D_i^{(s+1)} \) in Step 2 is a mixed \( \ell_1 \) and \( \ell_2 \) norm minimization problem with equality constraints; therefore it is difficult to find an optimal solution though it is a convex problem. Hence this paper proposes the following update schemes instead of Step 2.

**Step 2-1.**

\[
\{D_i^{(s+\frac{1}{2})}\}_{i=1}^L \leftarrow \text{argmin}_{\{D_i\}_{i=1}^L} \|D_iX^{(0)}\|_F^2 \quad \text{subject to } D_i \in \mathcal{D}
\]

**Step 2-2.**

\[
D_i^{(s+\frac{1}{2})} \leftarrow \max(0, D_i^{(s+\frac{1}{2})} - \gamma E) \text{ for all } i
\]

**Step 2-3.**

\[
\{X_i^{(s+\frac{1}{2})}\}_{i=1}^L \leftarrow \text{argmin}_{\{X_i\}_{i=1}^L} \|D_i - D_i^{(s+\frac{1}{2})}\|_F^2 + \tau \|V_d(D_i)\|_2^2
\]

**Step 2-4.**

\[
D_i^{(s+1)} \leftarrow D_i^{(s+\frac{1}{2})} \left( \sum_k D_k^{(s+\frac{1}{2})} \right)^{-1}, \text{ for all } i
\]

The above algorithm is based on the gradient projection method. At each step, \( D_i \) is updated for each of the objective function \( f_i \). Step 2-1 calculates the diagonal matrix \( D_i \) that minimizes \( \|Z - DX\|_F^2 \) by the least squares method as follows,

\[
(D_i)_{jj} = \frac{1}{L} \left( 1 - \frac{1}{N} \sum_{l=1}^L \frac{(X_j^T(Z_l))}{(X_j^T(X_l))} + \frac{(X_j^T(Z_l))}{(X_j^T(X_l))} + \frac{(X_j^T(Z_l))}{(X_j^T(X_l))} \right),
\]

where \((\cdot)\) denotes the \( j \)-th column vector of the matrix.

Step 2-2 corresponds to the third term of \( f_i \) and calculates sparse \( D_i \) by using a soft threshold. Step 2-3 regularizes the fourth term of \( f_i \), giving \( D_i^{(s+\frac{1}{2})} \) as,

\[
D_i^{(s+\frac{1}{2})} = \text{diag}(\sum_k(D_i^{(s+\frac{1}{2})} . \text{diag}(D_i^{(s+\frac{1}{2})})).
\]

Because \( D_i \) does not usually satisfy \( \sum_i D_i = E \) after the update of Step 2-3, \( D_i \) is projected to satisfy the equation in Step 2-4. Since Step 2-1, Step 2-2 and 2-3 update for each \( D_i \) while ignoring the constraints, parallel computation is possible in these steps. Since elements of \( D_i \) are non-negative at every step, the majority are assigned values between zero and one. In Step 3, \( X^{(s+1)} \) can be obtained as \( X^{(s+1)} = \mathcal{P}(Z, t_\epsilon, t_{\epsilon+1}) \), where \( \mathcal{P} \) denotes the projection operator on the constraint set \( \mathcal{H} \cap \mathcal{J} \). Let \( \mathcal{J}_{i,j} \) denote the index set of \( X_{k,l} \) corresponding to \( x_{i,j} = 1 \). Accordingly, it holds that \( X_{i,j} \) takes the same values for all \((k,l) \in \mathcal{J}_{i,j} \) for each \((i,j)\), since \( X \in \mathcal{H} \). From the least squares solution, \( \mathcal{P} \) can be simply computed, and \( X = \mathcal{P}(Z) \) is obtained as follows,

\[
X_{k,l} = \max(\min(\tau_{i,l}, 1), 0), \text{ if } (k,l) \in \mathcal{J}_{i,j}, \quad \tau_{i,l} = \begin{cases} 1 \sum_{(k,l) \in \mathcal{J}_{i,j}} \frac{Z_{k,l}}{H_{i,l}} & (i,j) \in \Omega^c \\ \frac{1}{H_{i,l}} & (i,j) \in \Omega \end{cases}.
\]

Algorithm 1 is a summary of the proposed algorithm, where \( \mathcal{S}(\cdot) \) and \( \mathcal{S}^{-1}(\cdot) \) denote the transfer operators from image \( I \) to structured matrix \( X \) defined in (3), and its inverse, respectively. The solution obtained by the proposed algorithm depends on the initial value of \( D_1 \). Experimentally, using the ALOHA algorithm [14], a good scheme involves setting \( D_1^{(0)} \) and \( L \) using mean-shift clustering. The computational complexity of Algorithm 1 depends on Step 1 and 2. SVD of \( D_iX \in R^{M-2K(N-2K)\times(2K+1)^2} \) requires \((2K+1)^2(M-2K)(N-2K)\) multiplications if \((M-2K)(N-2K) > (2K+1)^2\) is satisfied, moreover, the least square method used to solve diagonal matrices \( D_i \) requires \( O(MN^2) \) in Step 2-1 and 2-3. Therefore, the computational complexity of the algorithm is \( O(LK^2MN + (MN^2)) \) for each iteration.
Table 1: Average performance of algorithms (text mask).

|                | TV   | TV-H⁻¹ [22] | Exemplar | Low rank opt. [13] | ALOHA | Proposed algorithm |
|----------------|------|-------------|----------|--------------------|-------|--------------------|
| PSNR           | 24.94| 25.95       | 22.14    | 26.45              | 26.78 | 27.30              |
| SSIM           | 0.8735| 0.8865      | 0.8174   | 0.8872             | 0.9037| 0.9057             |

Table 2: Average performance of algorithms (80% random missing).

|                | TV   | TV-H⁻¹ [22] | Low rank opt. [13] | ALOHA | Proposed algorithm |
|----------------|------|-------------|--------------------|-------|--------------------|
| PSNR           | 23.11| 24.60       | 24.01              | 24.40 | 24.99              |
| SSIM           | 0.7578| 0.8083      | 0.7713             | 0.7899| 0.8132             |

Fig. 4: Results of inpainting for image number 95.
4. Numerical Examples

This section presents numerical examples. We used 100 RGB images with the dimensions of $300 \times 300$ pixels, which were obtained from an image database\(^1\) version 1.0.000. The proposed algorithm was applied to the RGB layers of the image independently. We used the parameters $K = 10$, $\lambda_0 = 0.05$, $\gamma = 0.05$, $\tau = 0.1$, $\alpha = 0.1$, $\varepsilon = 10^{-5}$ and $T_{\text{max}} = 150$ in Algorithm 1, with $T_{\text{max}}$ being set large enough to converge the algorithm. Two types of missing masks were used to inpaint the images by several methods. These are the dummy text mask shown in Fig. 3 and the random mask that randomly loses 80% of the pixels. In the objective evaluation, the proposed algorithm is compared with TV minimization, high order TV [22], exemplar

\(^1\)https://testimages.org/
Fig. 8 Results of inpainting for image number 74.

Based method [23], the structured rank minimization [13] and ALOHA [14].

Table 1 and 2 show the average of peak signal to noise ratio (PSNR) and structural similarity (SSIM) of the 100 images, from which it is evident that the proposed algorithm has the best performance out of all of the algorithms for both test conditions. In the case of random missing, the exemplar-based method was unable to determine blocks satisfying the conditions and could not perform image inpainting. Note that the average PSNR values of the approach without regularizations ($\tau = 0, \gamma = 0$) are 24.91 (text mask) and 24.27 (random mask), respectively. This implies that

Fig. 9 Visualized model selector using proposed algorithm for image number 74.
two kinds of regularization terms for $D_i$, which are the third and fourth terms in (16), improve the recovery accuracy of image inpainting.

To be sure, the proposed algorithm is not effective for all images. In the case of text masks, the PSNR of the proposed algorithm is better than the other algorithms, for 78 of the 100 images. Figure 4, 6, 8 and 10 show the results of restoring an image with text mask and random mask, and Fig. 5, 7, 9 and 11 show the visualized model selector matrices $D_i$. In particular, Fig. 4 and 6 show images where the proposed algorithm gives the best recovery results, and Fig. 8 and 10 show images where the conventional algorithm gives better results than the proposed algorithm. Table 3 shows the PSNR of each image, from which it is evident that the proposed algorithm obtains the highest PSNR for image numbers 95 and 39, and the ALOHA algorithm has a better performance than the other algorithms for image numbers 74 and 36. Indeed, the model selectors of the proposed algorithm for image number 95 and 39 divided each object area effectively. Conversely, for image number 74, the proposed algorithm results in excessive scattering and meaningless segmentation and, in image number 36, the segmentation is confused between the sky and the ramp. In light of this information, it is clear that whether the proposed algorithm is effective or not depends on whether clustering is successful.

Next we focus on the computing time. The experiments were conducted on a personal computer using Intel Core i7-6950X with 32 Gbytes RAM, and the algorithms were implemented in MATLAB. Computing time is compared in Table 4. We can see that the computing time of the proposed algorithm depends on the value of $L$ and is in the same order of ALOHA. Note that, the proposed algorithm converges at $t < T_{\text{max}}$ for all images.

Next, we compare the restoration quality of the proposed algorithm with that of the structured matrix rank minimization algorithm [13] using a number of block sizes for block division. Table 5 shows the PSNR values of the results of algorithm [13] and the proposed algorithm. The block division is implemented as square block division that is half overlapped with no window function. The restoration quality of the block division approach with the matrix rank minimization based algorithm depends strongly on the selected
Table 3  PSNR of each image.

| Number of images | TV       | TV-\(H^{-1}\) [22] | Exemplar | Low rank opt. [13] | ALOHA | Proposed algorithm |
|------------------|----------|---------------------|----------|-------------------|-------|-------------------|
| 95               | 20.74    | 22.00               | 18.41    | 21.92             | 21.62 | **23.05**         |
| 39               | 23.72    | 24.56               | NaN      | 24.66             | 25.71 | **25.98**         |
| 74               | 29.08    | 29.63               | 25.42    | 30.29             | **33.41** | 31.82 |
| 36               | 28.78    | 31.56               | NaN      | 29.86             | **32.51** | 30.71 |

Table 4  Average computing time [sec] comparison of algorithms.

| Missing condition | TV minimization | TV-\(H^{-1}\) [22] | Exemplar | Low rank opt. [13] | ALOHA | Proposed algorithm |
|-------------------|----------------|---------------------|----------|-------------------|-------|-------------------|
| text mask         | 2.75 \times 10^3 | 7.91 \times 10^3    | 6.12 \times 10^3 | 1.98 \times 10^3 | 3.36 \times 10^3 | 3.24 \times 10^3 |
| random (80%)      | 4.85 \times 10^3 | 1.12 \times 10^3    | NaN      | 1.60 \times 10^3  | 1.91 \times 10^3 | 2.30 \times 10^3 |

Table 5  Performance comparison of the proposed algorithm and rank minimization based algorithm with several block sizes.

| PSNR              | rank minimization algorithm with block division | proposed algorithm |
|-------------------|-----------------------------------------------|--------------------|
| Missing condition | block size 30 × 30 | 50 × 50 | 80 × 80 | 100 × 100 | 30 × 30 | 50 × 50 | 80 × 80 | 100 × 100 | 30 × 30 | 50 × 50 | 80 × 80 | 100 × 100 |
| text mask         | 24.36 | 26.24 | 26.83 | 26.87 | 27.30 |
| random (80%)      | 22.48 | 24.17 | 24.51 | 24.34 | 24.99 |

Fig. 12  Results of large missing case with exemplar based algorithm and proposed algorithm.

block size, whereas the proposed algorithm performs well without specifying an optimal block size.

Finally, we discuss the case with large missing pixels. Figure 12 shows the results of the ALOHA algorithm, the proposed algorithm, and the exemplar algorithm [23] when the missing area is large. As mentioned in the introduction, exemplar based approaches recover missing pixels effectively when the missing region is large. From Fig. 12, it is evident that the proposed algorithm has not improved the restoration result from the initial value.

5. Conclusion

This paper proposed a multiple AR model approach to image inpainting. Indeed, algorithms based on position-invariant linear modeling and the block algorithm with a fixed block size are insufficient for restoring natural photographs. To improve inpainting quality, this work proposed a new multiple matrix rank minimization approach with sparse regularization. Numerical examples suggested that the results of the proposed algorithm are better than those of algorithms based on single invariant models.

Acknowledgments

This work was supported by JSPS KAKENHI Grant Number JP19K20303.

References

[1] W. Fan, J.M. Chen, and W. Ju, “A pixel missing patch inpainting method for remote sensing image,” 2011 19th International Conference on Geoinformatics, pp.1–4, June 2011.

[2] A. Criminisi, P. Perez, and K. Toyama, “Region filling and object removal by exemplar-based image inpainting,” IEEE Trans. Image Process., vol.3, no.9, pp.1200–1212, 2004.

[3] Y. Wexler, E. Shechtman, and M. Irani, “Space-time video completion,” Proc. Computer Vision and Pattern Recognition, vol.1, pp.120–127, 2004.

[4] Y. Liu and V. Caselles, “Exemplar-based image inpainting using multiscale graph cuts,” IEEE Transactions on Image Processing, vol.22, no.5, pp.1699–1711, May 2013.

[5] J. Wang, K. Lu, D. Pan, N. He, and B.-K. Bao, “Robust object removal with an exemplar-based image inpainting approach,” Neurocomputing, vol.123, pp.150–155, 2014.

[6] T. Amano and Y. Sato, “Image interpolation using hbp method on the eigenspace,” Systems and Computers in Japan, vol.38, no.1, pp.87–96, 2007.

[7] T. Ogawa and M. Haseyama, “Adaptive subspace-based inverse projections via division into multiple sub-problems for missing image data restoration,” IEEE Trans. Image Process., vol.25, no.12, pp.5971–5986, 2016.

[8] R.N.J. Veldhuis, “Adaptive restoration of unknown pixels in digital images,” Signal Processing 4: Theories and Applications, pp.1469–1472, 1988.

[9] M. Bertalmio, A.L. Bertozzi, and G. Sapiro, “Navier-Stokes, Fluid Dynamics, and Image and Video Inpainting,” Proc. Computer Vision
Tomohiro Takahashi received the B.S. degree in the Department of Electrical, Electronics and Computer Engineering from Chiba Institute of Technology, Chiba, Japan, in 2010, the M.S. and the Ph.D. degrees in the Department of Management Science from Tokyo University of Science, Tokyo, Japan, in 2012 to 2015, respectively. He was an Assistant Professor of Department of Information and Computer Technology at Tokyo University of Science. His current research interests include the field of signal processing, which includes image processing and audio signal processing. He is a member of IEEE, IEICE and IIEE.

Katsunori Uruma received the B.S., M.S. and Ph.D. degrees in the Department of Management Science from Tokyo University of Science, Tokyo, Japan, in 2012, 2014 and 2017, respectively. He was a junior research associate and an Assistant Professor, and in Fukuoka Institute of Technology form 1998 to 2001, a as- sistant at the Institue of Physical and Chemical Research (RIKEN) from 1998 to 2001, a as- sistant Professor at Department of Management Science, Faculty of Engineering, Science Uni- versity of Tokyo, from 2001 to 2002, an assis- tant Professor at Information Media Center of Hiroshima University, from 2002 to 2004, and an Associate Professor of the Department of Computer Science at Kogakuin University from 2004 to 2018. He is currently a Professor of Hosei University in 2018. His research interests are in matrix completion, matrix rank minimization, sparse optimization, signal restoration and system identification. He is a member of IEEE.

Toshihiro Furukawa received the B.S. and Ph.D. degrees from the Tokyo Institute of Techn- ology, Tokyo, Japan, in 1980 and 1990, respec- tively. He joined Canon Inc. in 1980, where he was engaged in the design of copy machines and optical disk system. He was involved in Chiba Institute of Technology from 1986 to 1992 as a research associate and an Assistant Professor, and in Fukuoka Institute of Technology form 1992 to 1999 as an Associate Professor, respectively. He is currently a Professor of the Department of Information and Computer Technology at Tokyo University of Science. His current research interests include adaptive signal processing, wireless communication methods, learning machine theory, and so on. He is a member of IEEE and IEICE.