Thai Journal of Mathematics
Special Issue (2022): Annual Meeting in Mathematics 2021
Pages 130 – 144
http://thaijmath.in.cmu.ac.th
ISSN 1686-0209

Image Reconstruction by Least-Squares Method using Adaptive Basis with Similarity Metric

Siriwan Intawichai and Saifon Chaturantabut*

Department of Mathematics and Statistics, Faculty of Science and Technology, Thammasat University, Pathumthani, 12120, Thailand
e-mail : siriwan_amth50@hotmail.com(S. Intawichai); saifon@mathstat.sci.tu.ac.th (S. Chaturantabut)

Abstract This work proposes a method for image reconstruction based on the least-squares (LS) method that uses adaptive basis generated from the remaining image data with Euclidean-distance-based similarity metric. To approximate missing image components, the LS problem is used with a low-dimensional basis that optimally captures the main features of the image. This basis can be constructed based on the singular value decomposition (SVD). The efficiency of this low-rank basis from SVD depends mainly on the selected remaining image data used in the basis construction. In this work, an image is divided into many 2-dimensional patches. Each incomplete patch is approximated by the LS method with the optimal basis constructed from some available patches that are selected by similarity metric based on Euclidean distance. As a result, different incomplete image patches can adaptively use different appropriate patches to construct a more accurate optimal basis when compared to the one constructed from all available patches. This work also considers the randomized SVD (rSVD) to accelerate the computation of the optimal basis used in the approximation. The efficiency of the the proposed reconstruction technique is shown through the numerical experiments on images with different features. The computational time of the proposed method is compared for the cases when the low-dimensional bases are computed from SVD and rSVD.

MSC: 65D18; 65F55; 49M27

Keywords: Missing data reconstruction; Singular value decomposition; Randomized SVD; Patch grouping; Least-squares approximation

1. Introduction

Let X be an image in gray-scale level. The basic principle of linear image representation X can be written as

$$X = \sum_{i=1}^{N} a_i \phi_i, \tag{1.1}$$

where ϕ_i are the basis functions and a_i are the representation coefficients of the image X for $i = 1, 2, ..., N$.

*Corresponding author.

The basis $\{\phi_i\}_{i=1}^{N}$ is an essential ingredient in the image representation and has been investigated by many works. The basis used for image representation can be separated into two categories. One is directional basis, such as curvelet basis, wavelet basis, and contourlet basis. Another is adaptive basis, which is constructed by adapting its content to fit or represent all characteristic of an images then it has better performance than the other. One of the most important methods for image representation with optimal basis selection is the singular value decomposition (SVD).

SVD is a standard techniques that has been used extensively for matrix decomposition [1–9]. It is an efficient method for extracting a set of orthonormal basis, which is optimal in the least-squares sense that it packs the maximum signal energy as few coefficients as possible [8]. SVD is very useful in many real-world applications such as low-rank matrix approximations [7, 8], image processing [10–13], image compression [11, 12], face recognition [12] and missing data image reconstruction [13–16].

It is well known that computing the full SVD may suffer from a heavy computational cost and storage requirement when the dimension of matrices grows larger which in turn limits their scalability. To deal with the high dimension matrix factorization, the methods for decreasing the computational work have been developed. Randomized singular value decomposition (rSVD) has been introduced by researchers from a wide range of areas, such as T. Sarlos [17] , E. Liberty et. al. [18] and N. Halko et al. [19], which obtained the rSVD algorithm in low-rank matrix approximations. The rSVD algorithm has been shown to extract low-rank approximations while still preserving the accuracy.

In the application of image reconstruction, S. Intawichai et. al [21, 22] aimed to reconstruct an image with missing pixels by extending the notion of least-squares (LS) approximation. Proper orthogonal decomposition (POD) can be used to generate an optimal low-dimensional basis, called POD basis, in Euclidean norm, which is essential in the LS method for applying to approximate the missing data. This approach also used the rSVD method to build the POD basis to decrease the computational times and reduce the reconstruction errors when compared to the SVD method. S. Intawichai et. al [23] introduced an image reconstruction approach by clipping a target image into many 2-dimensional patches, which would be separated into the sets of complete and corrupted patches. All complete patches were used to construct the POD basis, which would be later used with the LS approximation. The known pixels in the neighborhood around the missing components were used to reconstruct each incomplete patch. This approach also efficiently applied the rSVD method to generate the POD basis,which could further decrease the computation time with equivalent accuracy In this paper, we modify the image reconstruction approach in [23] by combining patch grouping to construct the adaptive basis. The patch grouping step specifies similar patches for each corrupted patch by using the Euclidean-distance-based similarity metric. The set of similar patches is then used to compute the adaptive basis using POD. In addition, we use SVD and rSVD methods to build this basis. This step can improve the performance of POD basis computation used in reconstruction approach.

The remaining parts of this work are organized as follows. Section 2 provides some background on SVD and rSVD. The proposed method for recovery the missing data is discussed in Section 3. In section 4, we perform numerical experiments that compare the proposed method with the method in [23], as well as investigate the effects of using SVD and rSVD for constructing POD basis. The conclusion is finally discussed in Section 5.

2. Singular Value Decomposition and Randomized Singular Value Decomposition

Let $X = [x_1, x_2, \dots, x_n] \in \mathbb{R}^{m\times n}$ be a full rank matrix, then suppose $rank(X) = r$ with $r = \min\{m, n\}$.

Definition 2.1. The singular value decomposition (SVD) of X is

$$X = U\Sigma V^T = \sum_{i=1}^{r} \sigma_i u_i v_i^T, \tag{2.1}$$

where $U \in \mathbb{R}^{m\times m}$ and $V \in \mathbb{R}^{n\times n}$ are matrices which the column vectors are orthonormal. The matrix $\Sigma = diag(\sigma_1, ..., \sigma_r) \in \mathbb{R}^{m\times n}, r = \min\{m, n\}$ where $\sigma_1 \geq \sigma_2 \geq ... \geq \sigma_r > 0$.

The vectors u_i and v_i, which are the column vectors of U and V, respectively, are called left and right singular vectors of X. It can be shown that u_i and v_i are eigenvectors of XX^T and X^TX, respectively. In addition, $\sigma_1, \sigma_2, ..., \sigma_r$ are known as singular values. Note that, computing SVD of X directly can be time consuming, especially for large m and n. In this case, if $m \gg n$, it is more efficient to compute SVD by applying the method of snapshots, which is based on computing the eigenvalue decomposition of a smaller matrix X^TX as shown in Algorithm 1.

Algorithm 1: The SVD algorithm

Input	: A data matrix $X \in \mathbb{R}^{m\times n}$.
Step 1.	Set $B = X^TX$
Step 2.	Find the eigendecomposition of B, $B = WDW^T$
Step 3.	Sort D in descending order, $D1 = sort(D)$
	Set V by the vectors correspoding to $D1$, $V = cor(W)$
Step 4.	Calculating $\Sigma = \sqrt{D1}$
Step 5.	Computing $U = XV\Sigma^{-1}$
Output	: The SVD of X: U, Σ, V

In many applications, we can reduce computational cost by dealing with a low-rank matrix approximation of a data matrix X, which can be obtained from the truncated SVD. Truncated SVD is constructed from the full SVD by choosing the top k dominant singular values and their corresponding singular vectors as described in the following definition.

Definition 2.2. Truncated SVD of X is

$$X_k = U_k\Sigma_k V_k^T = \sum_{i=1}^{k} \sigma_i u_i v_i^T, \tag{2.2}$$

where $k < r$ is the numerical rank, $U_k \in \mathbb{R}^{m\times k}$ and $V_k \in \mathbb{R}^{n\times k}$ are matrices with orthonormal columns $\Sigma_k = diag(\sigma_1, ..., \sigma_k) \in \mathbb{R}^{k\times k}$ where $\sigma_1 \geq \sigma_2 \geq ... \geq \sigma_k > 0$.

The minimum errors in the Frobinius norm and 2-norm of the optimal k-rank approximation X_k to the matrix X given in the above definition are provided in the following lemma.

Lemma 2.3. *A low-rank matrix $X_k = [\widetilde{x}_1, \widetilde{x}_2, \ldots, \widetilde{x}_n]$ is the best rank k approximation for the matrix X with the low-rank matrix approximation error measured by 2-norm:*

$$\|X - X_k\|_2^2 = \sigma_{k+1}^2, \qquad \textit{and} \qquad \|X - X_k\|_F^2 = \sum_{i=1}^{n} \|x_i - \widetilde{x}_i\|_2^2 = \sum_{\ell=k+1}^{r} \sigma_\ell^2. \quad (2.3)$$

Next, we will define the randomized singular value decomposition (rSVD). The rSVD of X can be formed as .

$$\widehat{X}_k \approx \widehat{U}_k \widehat{\Sigma}_k \widehat{V}_k^T, \quad (2.4)$$

where $k < r$, $\widehat{U}_k \in \mathbb{R}^{m \times k}$ and $\widehat{V}_k \in \mathbb{R}^{n \times k}$ are matrices with orthonormal columns. $\widehat{\Sigma}_k \in \mathbb{R}^{k \times k}$ is a diagonal matrix whose diagonal entries are the singular values $\widehat{\sigma}_1 \geq \widehat{\sigma}_2 \geq \ldots \geq \widehat{\sigma}_k > 0$. Note that, the matrices $\widehat{U}_k$, $\widehat{\Sigma}_k$, and $\widehat{V}_k$ from rSVD are obtained from Algorithm 2 and generally different from the standard SVD.

To construct rSVD, define the random projection of a matrix X as

$$A = X\Omega, \quad (2.5)$$

where Ω is a random matrix of dimension n-by-$(k+p)$, k is the target low rank and p is a small positive integer used as an oversampling parameter that can be chosen arbitrarily. The rSVD method was introduced in [19] to reduce the computational time in approximating matrix factorizations by using random projections. The procedure for computing rSVD is separated into two stages. First, the random sampling matrix is applied to the data matrix X for generating a reduced matrix whose range still approximates the range of this original data matrix. Next, the resulting reduced matrix is factorized using SVD. Algorithm 2 provides details of the procedure for constructing the rSVD of a given data matrix.

Algorithm 2: The rSVD algorithm

Input	: A data matrix $X \in \mathbb{R}^{m \times n}$ with target rank k and an oversampling p
Stage A.	
Step 1.	Define a random matrix Ω with dimension $n \times (k+p)$
Step 2.	Compute the matrix product $A = X\Omega$
Step 3.	Construct Q by computing the QR decomposition of A
Step 4.	Set a reduce matrix, $B = Q^T X$
Stage B.	
Step 5.	Compute the SVD of B, $B = \widetilde{U}\widehat{\Sigma}\widehat{V}^T$
Step 6.	Set $\widehat{U} = Q\widetilde{U}$
Output	: The rSVD of X: $\widehat{U}, \widehat{\Sigma}, \widehat{V}$

3. Proposed Method

We modify an image reconstruction approach in [23] by combining patch grouping with an adaptive basis based on similarity metric. Figure 1 shows a stage-diagram of the proposed method. To design the adaptive basis, we begin with the patch grouping step. This step specifies similar patches of a target patch by the Euclidean-distance-based similarity metric. This target patch can be a corrupted patch that we want to reconstruct. In the next step, an adaptive basis is constructed by the SVD or rSVD based on the selected patches obtained in the previous step. This basis is then used in the reconstruction of the corrupted patch.

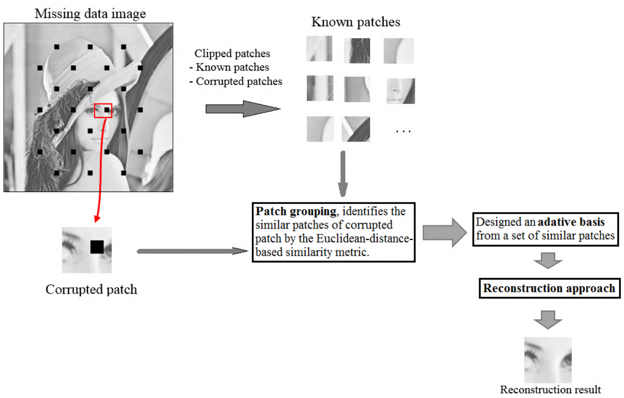


Figure 1. The stage-diagram of the proposed method for reconstruct the missing data in a damaged image.

Let X be a gray scale image. We begin with clipping the image X to the $\sqrt{m} \times \sqrt{m}$ patches. We then get n small patches with size $\sqrt{m} \times \sqrt{m}$ pixels and the patches can be separated to two sets, that is, S_c and S_{in}. Let $S_c = \{s_1, s_2, \dots, s_{n_c}\} \in \mathbb{R}^n$ and $S_{in} = \{\widehat{s}_1, \widehat{s}_2, \dots, \widehat{s}_{n_{in}}\} \in \mathbb{R}^n$ be the sets of the complete and corrupted patches, respectively, whose the elements in both sets are vectorized from 2D matrices to be vectors in $\mathbb{R}^n$.

Suppose that $\widehat{s} \subset \mathbb{R}^n$ is an incomplete vector which represents the corrupted patch with $n = n_f + n_g$ components where n_f and n_g are the numbers of known and unknown components respectively. We will estimate the adaptive basis which is suitable for reconstructing each corrupted patch by combining the patch grouping with POD basis.

Patch Grouping

Based on the patch grouping strategy in [20], for each incomplete vector $\widehat{s} \in \mathbb{R}^n$, we find its similar vectors from S_c by the Euclidean-distance-based similarity metric, which is defined by

$$Sim(\widehat{s}, s_c) = \|\widehat{s} - s_c\|_2^2, \tag{3.1}$$

where $\|.\|_2$ denotes the Euclidean-distance and $s_c \in S_c$ is a candidate patch. The smaller the value of $Sim(\widehat{s}, s_c)$ is, the more similar $\widehat{s}$ and s_c are. The group of its L- most similar patches denoted by $\{s_{c,i}\}_{i=1}^{L}$ are chosen to construct the adaptive basis by using SVD or rSVD methods.

Reconstruction

Recall that, $\widehat{s}$ is the incomplete vector that has $n = n_f + n_g$ elements. Now, we need to factorize $\widehat{s}$ to the vectors of known and unknown components. Let $\{f_1, f_2, \dots, f_{n_f}\}$, $\{g_1, g_2, \dots, g_{n_g}\} \subset \{1, 2, \dots, n\}$ be the indices of the known and unknown components, respectively, of $\widehat{s}$. Suppose that $F = [e_{f_1}, \dots, e_{f_{n_f}}] \in \mathbb{R}^{n \times n_f}$ and $G = [e_{g_1}, \dots, e_{g_{n_g}}] \in \mathbb{R}^{n \times n_g}$ are the matrices which the columns $e_{f_i}, e_{g_i} \in \mathbb{R}^n$ are the f_i-th, g_i-th column of the identity matrix I_n, respectively. Let $\widehat{s}_f := F^T \widehat{s} \in \mathbb{R}^{n_f}$ and $\widehat{s}_g := G^T \widehat{s} \in \mathbb{R}^{n_g}$. Then, the known and the unknown components are separately contained in the vectors $\widehat{s}_f$ and $\widehat{s}_g$, respectively. Note that, when F^T is pre-multiplied, it extracts the n_f rows corresponding to the indices $f_1, \ldots, f_{n_f}$. Similarly, pre-multipying G^T is equivalent to extracting the n_g rows corresponding to the indices $g_1, \ldots, g_{n_g}$.

To approximate the missing elements contained in $\widehat{s}_g$, we fist project $\widehat{s}$ onto the column span of adaptive basis matrix U with $k = rank(U)$. This basis is constructed from the set of similar patches of $\widehat{s}$ together with using the SVD or rSVD method. That is,

$$\widehat{s} \approx Ua, \quad \text{or} \quad \widehat{s}_f \approx U_f a \quad \text{and} \quad \widehat{s}_g \approx U_g a,$$

where $a \in \mathbb{R}^k$ is a coefficient vector, $U_f := F^T U \in \mathbb{R}^{n_f \times k}$, and $U_g := G^T U \in \mathbb{R}^{n_g \times k}$. The coefficient vector a can be specified by considering the known components contained in $\widehat{s}_f$ through the approximation $\widehat{s}_f \approx U_f a$, in the following least-squares problem:

$$\min_{a \in \mathbb{R}^k} \|\widehat{s}_f - U_f a\|_2^2. \tag{3.2}$$

The solution of the problem (3.2) is given by $a = U_f^\dagger \widehat{s}_f$, where $U_f^\dagger = (U_f^T U_c)^{-1} U_c^T$ is the Moore-Penrose psudoinverse. That is,

$$\widehat{s}_g \approx U_g a = U_g U_f^\dagger \widehat{s}_f. \tag{3.3}$$

Algorithm 3 describes the steps for missing data reconstruction using least-squares approach with similarity metric.

Algorithm 3: Missing data reconstruction via Least Squares approach

Input	: Complete data set $S_c \subset \mathbb{R}^n$ and a target rank k Incomplete data vector $\widehat{s} \in \mathbb{R}^n$ with known entries in $\widehat{s}_f \in \mathbb{R}^{n_f}$ and unknown entries in $\widehat{s}_g \in \mathbb{R}^{n_g}$, where $n = n_f + n_g$
Step 1.	Find the similar vectors of $\widehat{s}$, that is $\{s_i\}_{i=1}^L \in S_c$ and form $S = [s_1, \ldots, s_L] \in \mathbb{R}^{n \times L}$ and let $r = rank(S)$
Step 2.	Compute basis U of rank $k \leq r$ by using SVD or rSVD method
Step 3.	Solve the coefficient $a \in \mathbb{R}^k$ from the least Squares problem in (3.2): $\min_{a \in \mathbb{R}^k} \|\widehat{s}_{n_f} - U_f a\|_2^2$
Step 4.	Compute the approximation $\widehat{s}_g \approx U_g a$
Output	: Approximation of $\widehat{s}_g$

The Optimal Basis

In this work, the basis is considered by computing the singular value decomposition (SVD). This basis can be obtained from the left singular vectors as follows.

For a given matrix $X = [x_1, \ldots, x_n] \in \mathbb{R}^{m \times n}$ and $k < r = rank(X)$. The SVD of X is $X = U\Sigma V^T$, where $U = [u_1, \ldots, u_r] \in \mathbb{R}^{m \times r}$ and $V = [v_1, \ldots, v_r] \in \mathbb{R}^{n \times r}$ are matrices with orthonormal columns which the column vectors u_i and v_i are the left and right singular vectors, respectively. $\Sigma = diag(\sigma_1, \ldots, \sigma_r) \in \mathbb{R}^{r \times r}$ with $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0$. As discussed in Section 2, $X_k^* = U_k \Sigma_k V_k^T$ is the optimal solution of the least- squares problem

$$\min_{X_k} \|X - X_k\|_F^2, \;\; rank(X_k) = k \tag{3.4}$$

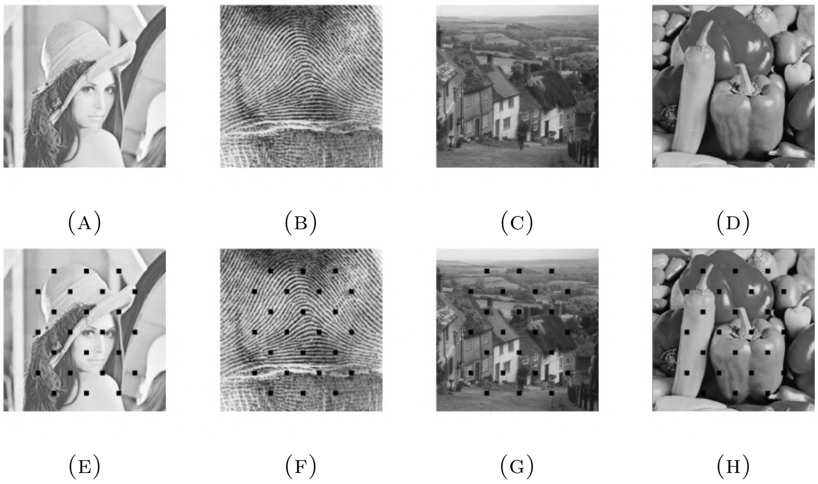


FIGURE 2. (A-D) The original gray-scale test images (E-H) Incomplete images with 2.06% missing pixels.

with minimum error $\|X - X_k^*\|_F^2 = \sum_{\ell=k+1}^{r} \sigma_\ell^2$ (by Lemma 2.3). The optimal orthonormal basis of rank k, which is also called proper orthogonal decomposition (POD) basis of dimension k, is the matrix formed by the first k column vectors of U, i.e. $U_k = [u_1, \dots, u_k] \in \mathbb{R}^{m\times k}$, $k \le r$.

When dealing with high dimensional images, computing SVD could computational intensive. To overcome this issue, the rSVD method [17, 19] described in Section 2 can be used to reduce the complexity for computing this POD basis in the reconstruction process.

4. Experimental Results

This section demonstrates numerical experiments of the proposed method and investigates between using SVD and rSVD methods for constructing the adaptive basis. In addition, we compare this proposed method to the previous method in [23]. We consider the CPU times and the reconstruction errors which is the relative error in 2-norm. Moreover, the reconstruction performances are considered in term of peak signal-to-noise ratio (PSNR).

In general, the color (RGB) image is generated from three systems of additive primary colors, which are red, green, and blue. In computation, each color represents the intensity value ranging from 0 to 1, which can be represented in gray-scale level. For the purpose of testing our image reconstruction approaches, we therefore only consider the R-image component and represent it as a gray-scale image. The same procedure can be applied to G-image and B-image. Here, we use four standard test images in gray-scale color with size 512×512 : **Lena**, **Fingerprint**, **Hill**, and **Peppers** as shown in Figure 2.

These images are damaged with 2.06% missing pixels and observed in two cases.

Case I: Splitting the test image to 1024 patches with size 16×16 pixels
Case II: Splitting the test image to 256 patches with size 32×32 pixels

For the first case, we have the matrix representing the image with dimension 256×1024. The number of complete and incomplete columns in this matrix are 934 and 90, respectively. In another case, we have the matrix representing the image with dimension 1024×256. The number of complete and incomplete columns in this matrix are 186 and 70, respectively. We investigate the computation of the adaptive basis by using SVD and rSVD.

To test the reconstruction approaches, we considers low dimensional matrix with rank $k = 10, 15, 20, 25, 30$. The four examples of reconstruction images with rank $k = 20$ are shown in Figure 3. The results are quite similar when compare between using SVD and rSVD methods for each Cases I and II. However, for a selected method for computing POD basis, the results from Cases I and II are quite different. Some missing pixels in Case I are not recovered accurately because some patches contain mostly corrupted area and the known pixels in neighborhood around the incomplete patch are not sufficient to perform the reconstruction. However, since the patches in Case II are relatively large when comparing with Case I, the number of the known pixels are generally enough for the reconstruction.

We compare the performance of different approaches by using the relative error from the image reconstructions and the computational time for constructing the adaptive basis, which are shown in Figure 4 and Figure 5, respectively. The results show that the adaptive basis computed by the rSVD method gives a little better performance than the basis computed by the SVD method. In Figure 4, the images (A), (B), (D), (E) and (H) show that the results when using the rSVD method is more accurate than when using the SVD method, while (C), (F) and (G) are better in some rank k. When we compare Cases I and II, the reconstructed images of **Lena** and **Peppers** are slightly more accurate in Case I, but **Fingerprint**, and **Hill** are more accurate in Case II.

The simulation times for constructing the basis sets in Figure 5 are shown to be slightly oscillating for different rank k when using rSVD, which may result from generating new random projection every time we perform a reconstruction. Figure 5 also illustrates that the rSVD method mostly uses less CPU time than SVD.

In addition, the comparisons between the proposed approach and the previous approach in [23] when using SVD and rSVD for Case I and Case II are shown in Figure 6 and Figure 7. The efficiency of these strategies are measured by the peak signal-to noise ratio (PSNR), as shown in Figure 6. In Case I, the proposed approach gives the better resulting images than the previous approach when using SVD method, while using rSVD gives similar results for both approaches. In Case II, the results of the proposed approach are similarly to the previous approach. Figure 7 shows the computational time for computing the adaptive basis which demonstrates that the propose method in both Case I and Case II uses less CPU times than the previous method when employing SVD, but uses similar CPU time when employing rSVD. In conclusion, from our numerical tests, the proposed method is more accurate than the previous method in [23] mainly for Case I when smaller patches are used, while both methods give the same order of accuracy for Case II when larger patches are used.

5. Conclusion

This work introduced an image reconstruction approach using least-squares (LS) method with adaptive basis constructed from similarity metric of available image pixels. A given

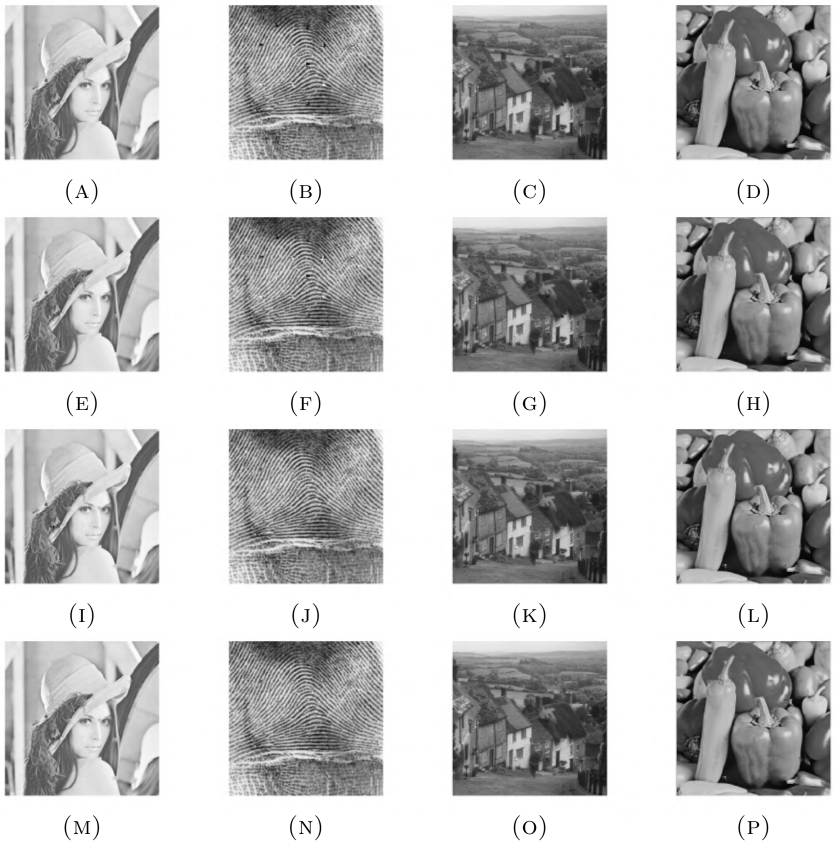
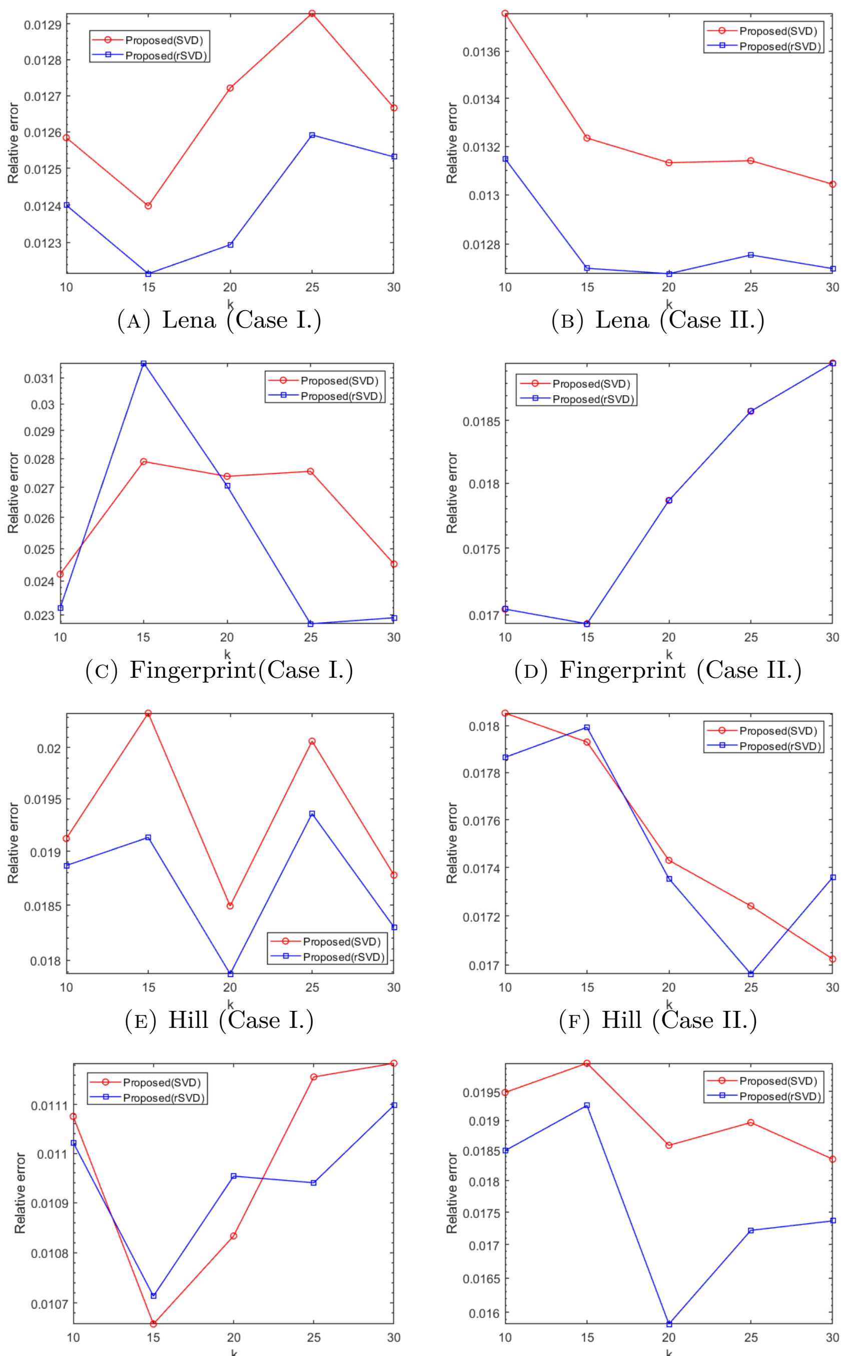


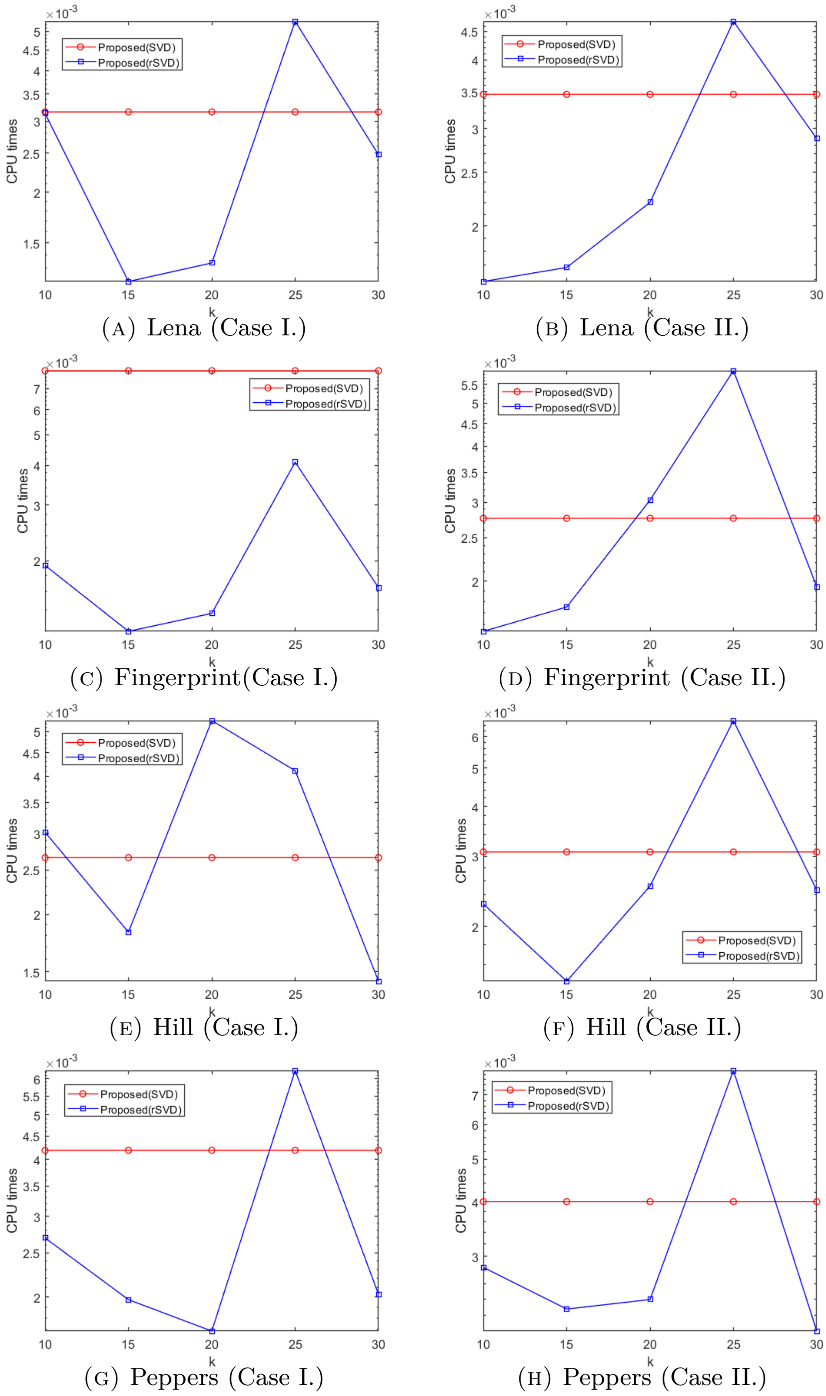
Figure 3. The reconstructed images when using the proposed method with rank $k = 20$. Case I: (A-D), (E-H) show the reconstructed images with the SVD and rSVD methods, respectively. Case II: (I-L), (M-P) show the reconstructed images with the SVD and rSVD methods, respectively.

test image with missing pixels was first divided into many patches. For each incomplete patch, some similar patches from the set of complete patches are selected by the Euclidean-distance-based similarity metric. The set of similar patches is then used to construct an adaptive basis for forming a low-dimensional subspace, which would be further used in the least-squares method to approximate the missing pixels in the incomplete patch. This work also applied the rSVD method to reduce the computational time of basis construction when compared with the standard SVD method. In the numerical experiments, we tested the proposed method on four images with different features. The results were shown to recover the missing pixels of these images accurately when compared with an existing method that does not use Euclidean-distance-based similarity metric.



(A) Lena (Case I.)
(B) Lena (Case II.)
(C) Fingerprint(Case I.)
(D) Fingerprint (Case II.)
(E) Hill (Case I.)
(F) Hill (Case II.)
(G) Peppers (Case I.)
(H) Peppers (Case II.)

FIGURE 4. The relative errors of the reconstructed images in Case I and Case II For **Lena**, **Fingerprint**, **Hill** and **Peppers**, respectively, when 2.06% of pixels are damaged.



(A) Lena (Case I.)

(B) Lena (Case II.)

(C) Fingerprint(Case I.)

(D) Fingerprint (Case II.)

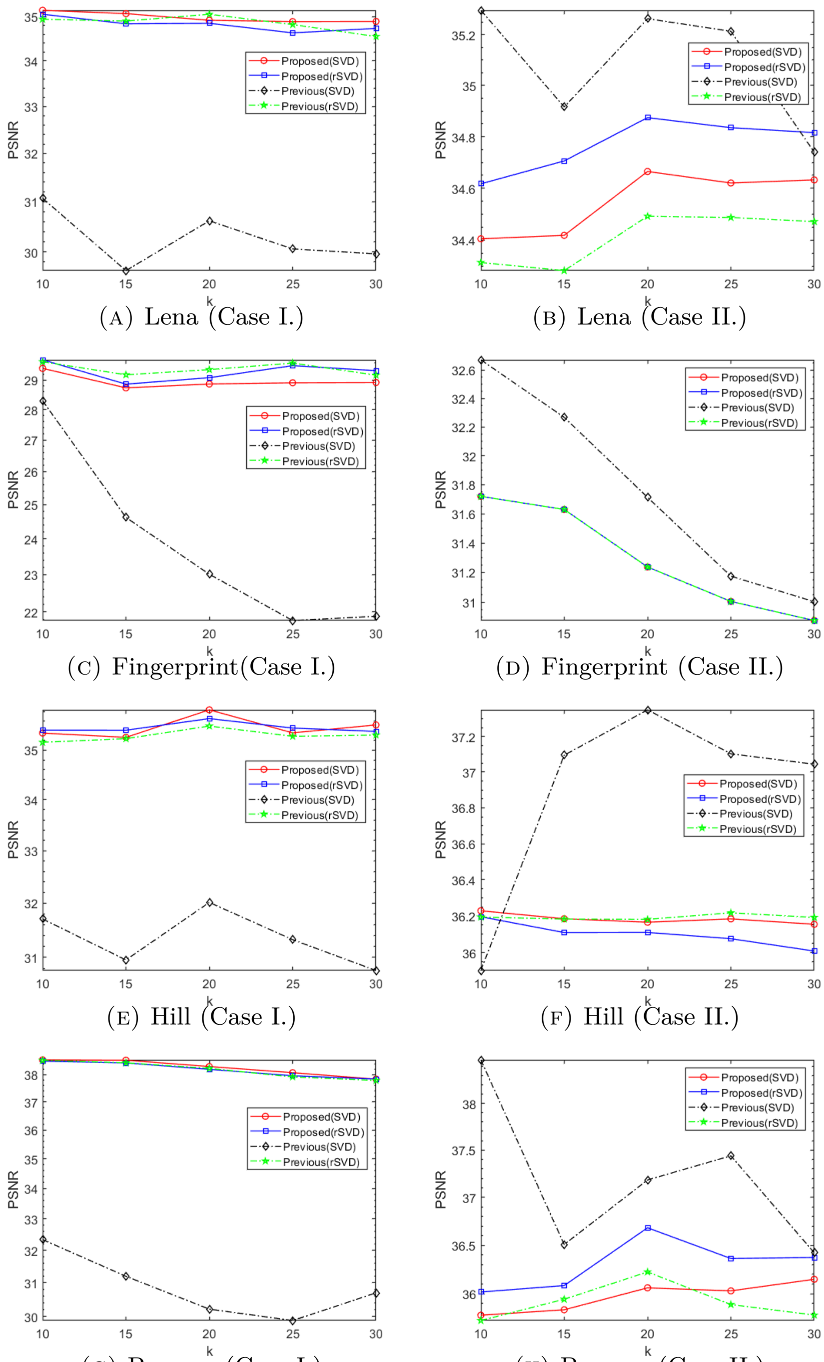
(E) Hill (Case I.)

(F) Hill (Case II.)

(G) Peppers (Case I.)

(H) Peppers (Case II.)

FIGURE 5. Computational time for computing the adaptive basis used in the reconstruction of Case I and Case II For **Lena**, **Fingerprint**, **Hill** and **Peppers**, respectively, when 2.06% of pixels are damaged.



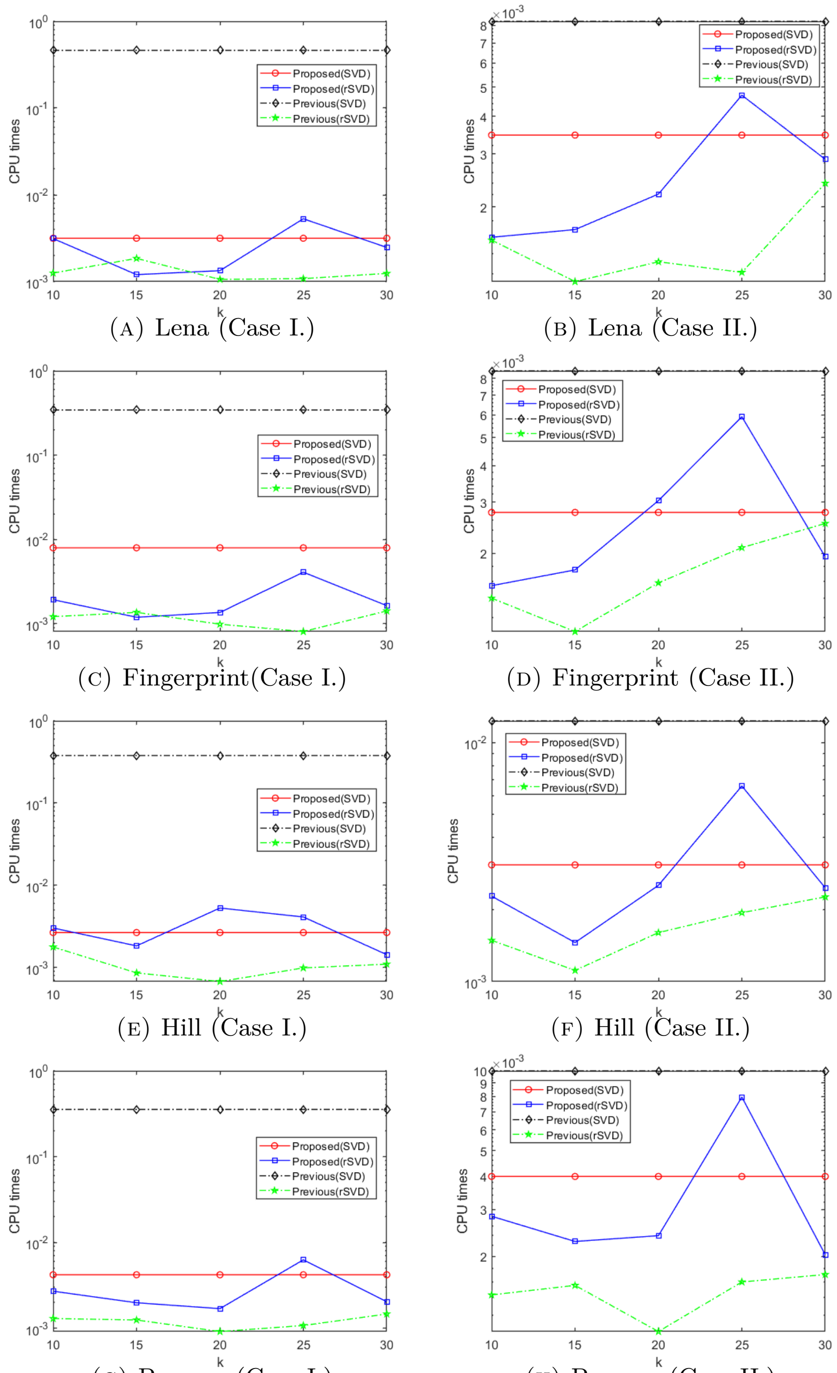
(A) Lena (Case I.) (B) Lena (Case II.)

(C) Fingerprint(Case I.) (D) Fingerprint (Case II.)

(E) Hill (Case I.) (F) Hill (Case II.)

(G) Peppers (Case I.) (H) Peppers (Case II.)

FIGURE 6. Reconstruction performances in term of PSNR of Case I and Case II of **Lena**, **Fingerprint**, **Hill** and **Peppers**, respectively, when compare to the previous method.



(A) Lena (Case I.) (B) Lena (Case II.)

(C) Fingerprint(Case I.) (D) Fingerprint (Case II.)

(E) Hill (Case I.) (F) Hill (Case II.)

(G) Peppers (Case I.) (H) Peppers (Case II.)

FIGURE 7. Computational time for computing the adaptive basis used in the reconstruction of Case I and Case II For **Lena**, **Fingerprint**, **Hill** and **Peppers**, respectively, when compare to the previous method.

Acknowledgements

The authors gratefully acknowledge the financial support provided by Thammasat University Research Fund, Contract No. TUFT 76/2564 and Royal Thai Government Scholarship in the Area of Science and Technology (Ministry of Science and Technology).

References

[1] G. H. Golub and C. Reinsch, Handbook series Linear Algebra: Singular value decomposition and least square solutions, Numer. Math. 4(1970), 403–420.

[2] V. C. Klema and A. J. Laub, The singular value decomposition: Its computation and some applications, IEEE Transactions on Automatic control. 25(1980), 164–176.

[3] J. Zhou, C. Kwan and B. Ayhan, A High Performance Missing Pixel Reconstruction Algorithm for Hyperspectral Images, Proc. 2nd International Conference on Applied and Theoretical Information Systems Research, (2012).

[4] F. Cao, M. Cai and Y. Tan, Image Interpolation via Low-Rank Matrix Completion and Recovery, IEEE Transactions on Circuits and Systems for Video Technology, 25(2015), 1261-1270.

[5] G. W. Stewary, On the early history of the singular value decomposition, Siam Review, 35(1993), 551-566.

[6] D. Kalman, A Singularly Valuable Decomposition: The SVD of a Matrix, The College Mathematics Journal 27(1996), 2-23.

[7] D. Achlioptas and F. McSherry, Fast computation of low-rank matrix approximations, J. Assoc. Comput. Mach. 54(2007).

[8] G. Strang, Linear Algebra and its Applications, 3rd ed., San Diego, USA: Harcourt Brace Jovanovich, 1988.

[9] G.H. Golub, C.F. Van Loan, Matrix Computations, 3rd ed., Johns Hopkins University Press, Baltimore, Maryland, 1996.

[10] H. C. Andrews and C. L. Patterson, Singular value decompositions and digital image processing 24(1976), 26-53.

[11] S. Kahu and R. Rahate, Image Compression using Singular Value Decomposition, International Journal of Advancements in Research & Technology 2(2013), 244-248.

[12] L. Cao, Singular Value Decomposition Applied To Digital Image Processing, Division of Computing Studies, Arizona State University Polytechnic Campus, Mesa, Arizona, 1-15.

[13] S. O. Aase, J. H. Husoy and P. Waldemar, A critique of SVD-based image coding systems, IEEE International Symposium on Circuits and Systems (ISCAS) 4(1999), 13-16.

[14] V. V. Selivanov and R. Lecomte, Fast PET Image Reconstruction Based on SVD Decomposition of the System Matrix, IEEE transactions on nuclear science 48(3)(2001), 761-767.

[15] R. A. Sadek, SVD Based Image Processing Applications: State of The Art, Contributions and Research Challenges, International Journal of Advanced Computer

Science and Applications 3(2012), 26-34.

[16] D. M. Lyra-Leite, J. P. C. L. da Costa and J. L. A. de Carvalho, Improved MRI reconstruction and denoising using SVD-based low-rank approximation, Workshop on Engineering Applications (2012), 1-6.

[17] T. Sarlos, Improved Approximation Algorithms for Large Matrices via Random Projections. In Foundations of Computer Science. 47th Annual IEEE Symposium (2006), 143-152.

[18] E. Liberty, F. Woolfe, P. G. Martinsson, V. Rokhlin, and M. Tygert, Randomized algorithms for the low-rank approximation of matrices, in Proc. Natl. Acad. Sci. 104(51)(2007), 20167-20172.

[19] N. Halko, P.G. Martinsson and J.A. Tropp, Finding Structure with Randomness: Probabilistic Algorithms for Constructing Approximate Matrix Decompositions, Siam Review 53(2011), 217-288.

[20] Q. guo, C.Zhang, Y.Zhang and H. Liu, An efficient SVD-based method for image denoising, IEEE transactions on circutts and systems for video technology, 26(2016), 868-870.

[21] S. Intawichai and S. Chaturantabut, An application of randomized singular value decomposition on image reconstruction using least-squares approach, Preprint in Thai Journal of Mathematics (2018).

[22] S. Intawichai and S. Chaturantabut, A Numerical Study of Efficient Sampling Strategies for Randomized Singular Value Decomposition, Thai Journal of Mathematics (2019), 351-370.

[23] S. Intawichai and S. Chaturantabut, A Missing Image Data Reconstruction based on Least-Squares Approach with Randomized SVD, Preprint in the Springer Nature: Advances in Intelligent Systems and Computing (2021).