Image Inpainting Based on Related Dictionary Constructed by Histogram

Qiaoqiao Li, Guoyue Chen, Xingguo Zhang, Kazuki Saruta, and Yuki Terata

***Abstract*—This paper proposed a new patch-wise image inpainting algorithm based on sparse representation with related dictionaries. For each target patch, i.e., the patch with corrupted parts, a related dictionary is defined, which consists of the patches in the uncorrupted area having a similar histogram with that of the patch. The related dictionary contains the most similar patches for target patch. That ensure the accuracy of the inpainting process. The target patch then can be inpainted by a sparse representation of the patches in the related dictionary. Experimental results are given to show the effectiveness of the proposed algorithm.**

***Index Terms*—Image inpainting, sparse representation, histogram, related dictionaries.**

I. Introduction

Image inpainting which is to fill the missing or corrupted area by using the known information of the image. These area may be individual missing pixels in the damage image, or be continuous regions resulting from man-made degradation and other reasons. Image inpainting is an important problem, because it has a wide variety of applications, such as image object removal, image restoration and transmission [1], [2]. In recently, it has attracted growing interest from researchers.

Generally, there are three types of image inpainting approaches, i.e.diffusion-based inpainting approach, exemplar-based inpainting approach, and inpainting approach based on sparse representation [2]. The diffusion-based inpainting approach is such that the corrupted area is restored by diffusing the surrounding information to the target region [3], [4]. However, these diffusion-based inpainting methods are not well suitable for the textured, especially if the target region to be restored is large [1]. The exemplar-based inpainting approach, which is inspired by the idea of texture synthesis technique [4], is that a patch in the corrupted area is first selected and then unknown pixels in the selected patch is filled by copying the pixels in the best matching patch in the whole source region by comparing the similarity of selected patch [1], [5]-[10]. Instead of using the best patch, inpainting approach based on sparse representation is to represent the image patches by using a sparse linear combination of atoms from a dictionary. Thus, the dictionary is very important for inpainting results. An effective sparse representation iterative inpainting algorithm is proposed [2]. And it is very suitable for recovering different structural components in the image. And the sparse representation can be adapted for restoring the color image [11]. This method is mainly extended from the denoising algorithm [12] whose dictionary is constructed by using the K-SVD [13]. After that, a similar algorithm is proposed in [14]. They directly used all the patches which are clipped from in the source region to construct a dictionary and obtained a good inpainting results. They analyzed their method is better than the algorithm [11]. Although the dictionary generated form all the patches in source region is advantageous to the image inpainting, it will have some of unrelated atoms with image patch to be restored. Moreover, the unrelated atoms will introduce the interference into inpainting results. As a result, it will affect the inpainting result.

The purpose of this paper is to propose a new inpainting algorithm based on sparse representation with related dictionaries. The proposed inpainting algorithm mainly consists of two parts. First, a related dictionary is constructed for each target patch, which is consists of the patches in the uncorrupted area, which have a similar histogram with that of the target patch. Second, the target patch is inpainted by a sparse representation of the patches in the related dictionary. Moreover, experimental results will be given to illustrate the effectiveness of the proposed inpainting algorithm.

This paper is organized as follows. Firstly, in Section II, we briefly describe sparse representation. Secondly, in Section III is devoted to describe an overview of the proposed algorithm based on sparse representation. Thirdly, in Section IV the experimental results are shown. The results demonstrate that the effective of the proposed method with dictionary preprocessed. Finally, in Section V, concludes this whole paper.

II. Sparse Representation

Reference [15] were first applied sparse signal representation to inpainting problems. Based on the hypothesis that natural signal can be represented by a linear combination of atoms in dictionary matrix.

Given a dictionary $\mathbf{D}$, each column of the dictionary $\mathbf{D}=[\mathbf{d_1},\mathbf{d_2},...,\mathbf{d_k}]$ can be named as an atom. Each target signal $\mathbf{y}=[y_1,...,y_k]^T$ can be represented as

$$\mathbf{y} \cong \mathbf{D}\boldsymbol{\alpha} \tag{1}$$

Manuscript June 18, 2018; revised August 30, 2018. This work was supported by the China Scholarship Council (scholarship 201708050084).

Qiaoqiao Li is with the Graduate School of Systems Science and Technology, Akita Prefectural University, Akita, 015-0055, Japan (e-mail: D18S005@akita-pu.ac.jp).

Guoyue Chen, Xingguo Zhang, Kazuki Saruta, and Yuki Terata are with the Dept. of Electronics and Information Systems, Akita Prefectural University, Akita, 015-0055, Japan (e-mail: chen@akita-pu.ac.jp, xingguozhang@akita-pu.ac.jp, saruta@akita-pu.ac.jp, tarata@akita-pu.ac.jp).

doi: 10.18178/ijmlc.2018.8.5.732

such that the coefficient vector $\boldsymbol{\alpha}$ have the least number of nonzero components. This is called the sparse representation problem and is usually cast as

$$\hat{\boldsymbol{\alpha}} = \operatorname{argmin}_{\boldsymbol{\alpha}} \|\mathbf{y} - \mathbf{D}\boldsymbol{\alpha}\|_2 \quad (2)$$

which can be solved by greedy algorithms, such as orthogonal matching pursuit (OMP) [16].

In image inpainting, each corrupted signal $\mathbf{y} = [y_1, y_2, 0, ..., y_k]^T$ where 0 denotes the corrupted entries. Removing the corrupted components can be used by

$$\mathbf{y}^* = \mathbf{M}\mathbf{y} \quad (3)$$

$$\mathbf{M} = \begin{bmatrix} 1 & & & & \\ & 1 & & & \\ & & 0 & & \\ & & & \ddots & \\ & & & & 1 \end{bmatrix} \quad (4)$$

$\mathbf{y}^*$ has all the uncorrupted entries of $\mathbf{y}$. Then, equation (1) can be rewritten as

$$\hat{\boldsymbol{\alpha}} = \operatorname{argmin}_{\boldsymbol{\alpha}} \|\mathbf{M}\mathbf{y} - \mathbf{D}\boldsymbol{\alpha}\|_2 \quad (5)$$

After obtaining the estimation of coefficients $\hat{\boldsymbol{\alpha}}$, we combination equation (1) with equation (5) and then the corrupted signal can be recovered by

$$\hat{y}_i = \begin{cases} (y^*)_i & if \ \ i \in \mathbf{y}^* \\ (\bar{\mathbf{M}}\mathbf{D}\boldsymbol{\alpha})_i & if \ \ i \notin \mathbf{y}^* \end{cases} \quad (6)$$

where $\bar{\mathbf{M}} = \mathbf{I} - \mathbf{M}$, $\mathbf{I}$ is a unit vector.

III. Algorithm

The overview of proposed method will be presented in this section in detail. The proposed method can be divided into three parts, find a target patch, generation of a dictionary and sparse reconstruction algorithm. In the first part, the target patch is found by computing the filling order. And the whole algorithm is starting from this patch. In the second part, we use the histogram to preprocess the original dictionary which is directly extracted from the original region. And then, we will obtain a related dictionary from the original dictionary. In the last part, we use the known information from the target patch to estimate their unknown information. Its details are shown in the following.

A. Filling Order

The filling order of image patches is crucial to inpainting results. In the inpainting methods, a different filling order can lead to different results [8]. In our method, we decide to use the filling order which is proposed in [7], since the structure information can be efficiently preserved. Recently, the filling order [7] has been used for their inpainting algorithms and obtained good results [14], [17]. Using the filling order to compute the every patch which is centered at p located the boundary δU between damaged region (target region $\mathbf{T}$) and undamaged region (source region $\mathbf{S}$). Finding the target patch $\boldsymbol{\psi}_\mathbf{p}$ has the maximum priority. As the patch $\boldsymbol{\psi}_\mathbf{p}$ is on the boundary, the patch contains the known pixels $\mathbf{A}$ and the unknown pixels $\mathbf{B}$. And we can see clearly the target patch in Fig. 1.

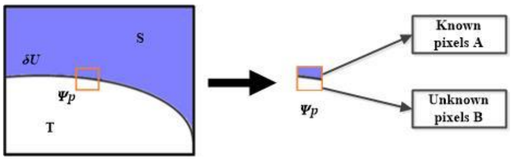


Fig. 1. A target patch with the maximum priority.

B. Generation of Dictionary

In order to compute the sparse representation of image patch, we first have to determine a dictionary. In the traditional method of generating dictionary, such as [14], directly use the whole patches clipped from the original region. A problem of it is that some unrelated patches can lead to some interference. So in the proposed method, we will preprocess the original dictionary by using histogram, as shown in Fig. 2.

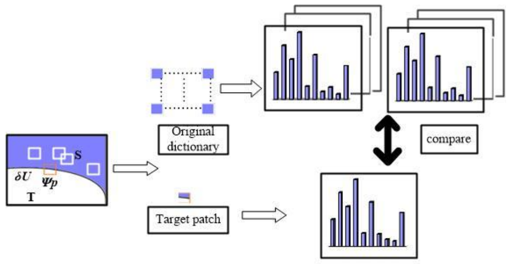


Fig. 2. Histogram comparison between target patch and original dictionary.

Let the histogram have N bins. Therefore, the histogram of color image have N values of the N bins of the histogram. For three channels R, G, B, the histogram of each channel can be represented by

$$\begin{aligned} \mathbf{histo}_\mathrm{R} &= [h_{\mathrm{R}1}, \cdots, h_{\mathrm{R}N}]^T \\ \mathbf{histo}_\mathrm{G} &= [h_{\mathrm{G}1}, \cdots, h_{\mathrm{G}N}]^T \\ \mathbf{histo}_\mathrm{B} &= [h_{\mathrm{B}1}, \cdots, h_{\mathrm{B}N}]^T \end{aligned} \quad (7)$$

The generation of the related dictionary can be divided into four steps:

1) Compute the histogram $\mathbf{HistoP} = [\mathbf{H}_1, \cdots, \mathbf{H}_m]^T$ of an original dictionary, which consists of the whole patches of original region. m is the number of the whole patches.
2) Find the target patch by using the filling order. After that compute the histogram of the target patch defined by

$$\mathbf{H}_\mathbf{T} = [\mathbf{histoT}_R, \mathbf{histoT}_G, \mathbf{histoT}_B] \quad (8)$$

3) Compare the histogram of the original dictionary with the histogram of target patch and find the similar patches. The following is a sample of comparing similarity between target patch and patch 1. We measure similarity using the histogram to via the following formula

$$\begin{aligned} simi &= \sum_{i=1}^{N} |\mathbf{H}_\mathbf{T}(\mathrm{i}) - \mathbf{H}_\mathbf{1}(\mathrm{i})| \\ &= \sum_{i=1}^{N} |\mathbf{histoT}_R(\mathrm{i}) - \mathbf{histoP}_R(\mathrm{i})| + \sum_{i=1}^{N} |\mathbf{histoT}_\mathrm{G}(\mathrm{i}) - \mathbf{histoP}_\mathrm{G}(\mathrm{i})| \\ &\quad + \sum_{i=1}^{N} |\mathbf{histoT}_\mathrm{B}(\mathrm{i}) - \mathbf{histoP}_\mathrm{B}(\mathrm{i})| \end{aligned} \quad (9)$$

where N is the number of bins, the similarity is to sum the difference of three channels.

The similarity between the whole patches of the dictionary and the target patch is given by

$$\mathbf{Simi} = [\mathrm{simi}_1, \cdots, \mathrm{simi}_m]^T \tag{10}$$

where m is the number of the patch, and the length of the vector Simi is also m.

In order to find the similar patches, we sort the similarity vector **Simi**

$$\mathbf{index}=\mathrm{sort}(\mathbf{Simi}) \tag{11}$$

and then we choose the top l patches as the similar patches.

4) Use the similar patches to generation the related dictionary, the atoms of the new dictionary have a large relevance to the target patch. This property ensures the inpainting result.

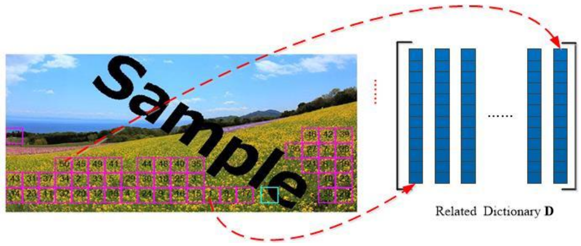


Fig. 3. Generate the related dictionary by using similar patches.

In the example shown in Fig. 3, the blue-green rectangle is denoted as target patch which is the patch we want to inpaint. The mulberry rectangle is denoted as the patches which we choose to generate the related dictionary. We choose the top l=50 similar patches $\left\{\boldsymbol{\psi}_{\mathbf{q}_j}\right\}_{j=1}^{50}$, among of them $\boldsymbol{\psi}_{\mathbf{q}_1}$ means the most similar one and it represents as the first column of the related dictionary. The rest patches can be done in the same manner.

C. Signal Recovery

After finding the patch $\boldsymbol{\psi}_{\mathbf{p}}$, the aim of the sparse reconstruction is using the known information $\mathbf{A}$ to estimate the unknown information $\mathbf{B}$. Matrix $\mathbf{M}$ has a special structure determined by the layout of the known pixels. Thus, the known information can be computed by

$$\mathbf{A} = \mathbf{M}\boldsymbol{\psi}_{\mathbf{p}} \tag{12}$$

As the target patch $\boldsymbol{\psi}_{\mathbf{p}}$ can be seen as the signal y, so according to (1), the (12) is redefined as

$$\mathbf{A} = \mathbf{MD}\boldsymbol{\alpha} \tag{13}$$

where $\mathbf{D}$ is the new related dictionary which is generated by comparing the histogram, and $\boldsymbol{\psi}_{\mathbf{p}} = \mathbf{D}\boldsymbol{\alpha}$. In the proposed method we will use the non-negative orthogonal matching pursuit (NNOMP) [18] algorithm which is an improved method of OMP, and then the estimation of sparse coefficients can be described by

$$\hat{\boldsymbol{\alpha}} = \arg\min_{\alpha \geq 0} \left\| \mathbf{A} - \mathbf{MD}\boldsymbol{\alpha} \right\|_2 \tag{14}$$

By this way, we only find out some non-negative coefficients. If we obtain the estimation of sparse coefficients $\hat{\boldsymbol{\alpha}}$ by using the NNOMP, the unknown information $\mathbf{B}$ can be restored approximately using

$$\mathbf{B} = \bar{\mathbf{M}}\mathbf{D}\hat{\boldsymbol{\alpha}} \tag{15}$$

where $\bar{\mathbf{M}} = \mathbf{E} - \mathbf{M}$ is a matrix which is decided by the layout of missing pixels and $\mathbf{E}$ is also a matrix with each entry being one. In detail, missing pixels in the target patch can be inpainted as follows:

$$\hat{\boldsymbol{\psi}}_p^i = \begin{cases} \boldsymbol{\psi}_p^i & if \;\; i \in \mathbf{A} \\ (\bar{\mathbf{M}}\mathbf{D}\hat{\boldsymbol{\alpha}})_i & if \;\; i \notin \mathbf{A} \end{cases} \tag{16}$$

As discussed above, the schematic diagram of we proposed inpainting algorithm is shown in Fig. 4.

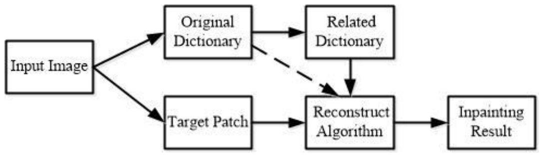
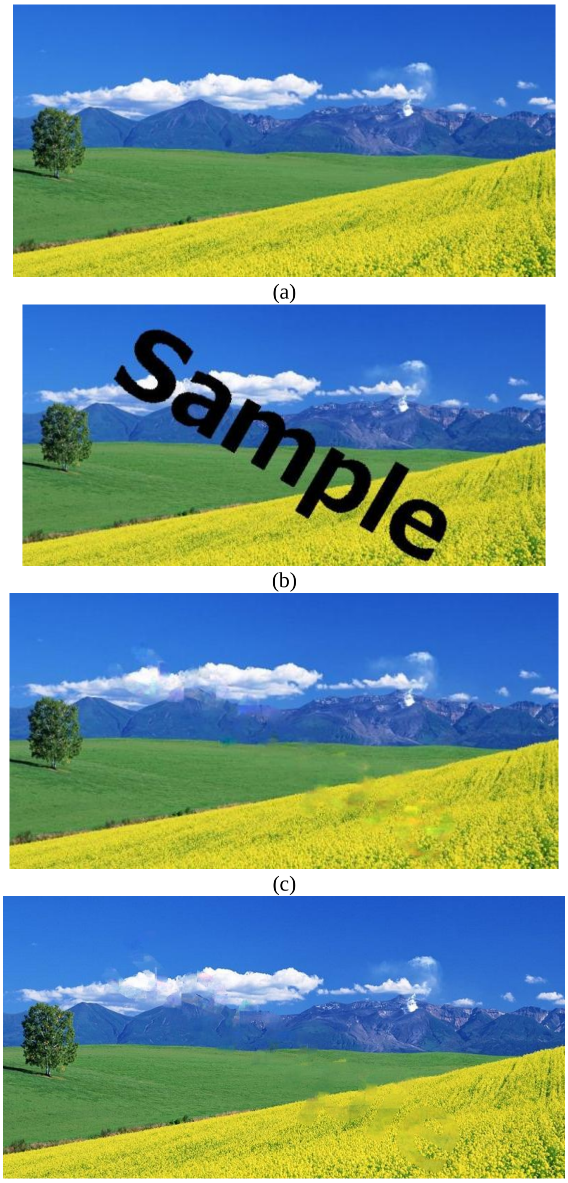


Fig. 4. Schematic diagram for the proposed algorithm.



(a)

(b)

(c)

(d)

Fig. 5. Inpainting result with different dictionaries: (a) Original image (b) corrupted image including test region (c) inpainting result obtained by using original dictionary (d) inpainting result obtained by using related dictionary.

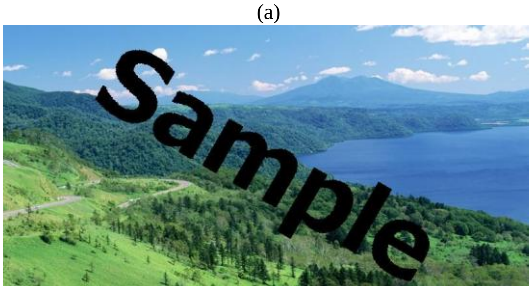
IV. Experiment Results

The experiment is conducted to demonstrate that generating related dictionary by using histogram is more effective than the original dictionary for image inpainting. In this experiment, we adopt different dictionaries to compare with each other.

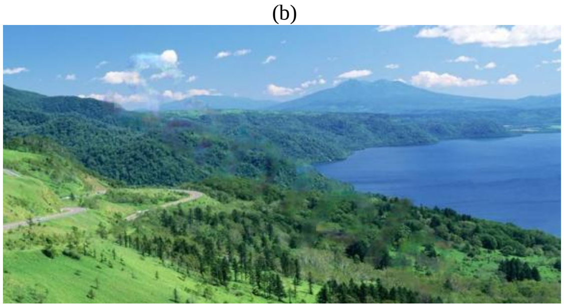
A. Image Inpainting with Different Dictionaries

In order to assess the performance of different dictionaries objectively, two dictionaries are used for comparison: original dictionary and related dictionary. Two pairs of testing images Fig. 5(a) and Fig. 5(b) show the original image and input images of one natural landscape, respectively. Fig. 6(a) and Fig. 6(b) show the original image and input images of another natural landscape, respectively.

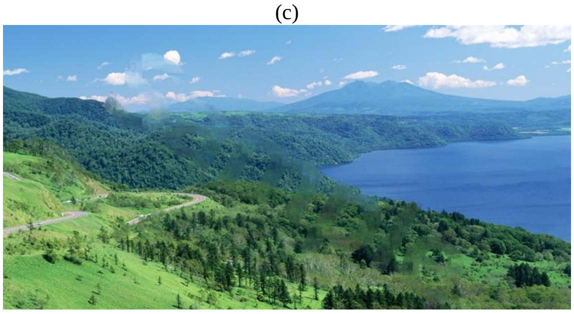
(a)



(b)



(c)



(d)

Fig. 6. Inpainting result with different dictionaries: (a) Original image(b) corrupted image including test region (c) inpainting result obtained by using original dictionary (d) inpainting result obtained by using related dictionary.

B. Evaluation Index

To measure the experimental results of quantitative evaluation by using Peak Signal-to-Noise Ratio (PSNR) obtained from MSE [19]. In addition, the values of PSNR in each color channel (R, G, B) are also computed in experiments. Different from computing mean square error (MSE) of the whole image, we just compute the mean square error of the defective region. Note that the mean square error is computed by using

$$\mathrm{MSE} = \frac{\sum_{(i,j)\in \mathrm{T}} (\mathrm{I}_{ori}(i,j) - \mathrm{I}_{inp}(i,j))}{\mathrm{N(T)}} \tag{17}$$

where $\mathrm{I}_{\mathrm{ori}}(i,j)$ is the brightness values of the original image and $\mathrm{I}_{inp}(i,j)$ is the brightness values of the inpainted image. T represents the target region, that is to say defective region, it is noted in Fig. 1. And $\mathrm{N(T)}$ represents the number of pixels in the defective region.

The PSNR is defined by

$$\mathrm{PSNR} = 10\log_{10}(\frac{255^2}{\mathrm{MSE}}) \tag{18}$$

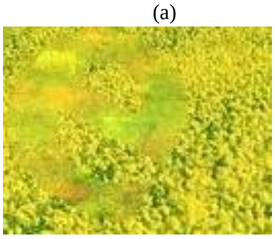
The experiment is conducted based on Fig. 4. And then we use the peak Peak Signal-to-Noise Ratio (PSNR) between original image and the inpainted image as the evaluation index to quantify the inpainting results. The results of quantitative metrics are depicted in Table I. From the Table I, we can learn that the proposed method by using the related dictionary has a better inpainting effect than the method by using the original dictionary.

TABLE I: Objective Performance

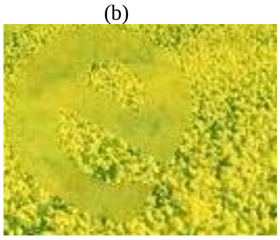
	RGB	Original dictionary	Related dictionary
Fig. 5	R	18.45	18.78
	G	20.58	21.01
	B	23.66	24.19
	RGB average	20.40	20.80
Fig. 6	R	18.28	18.79
	G	19.26	19.71
	B	20.25	20.91
	RGB average	19.19	19.72

(a)

(b)



(c)



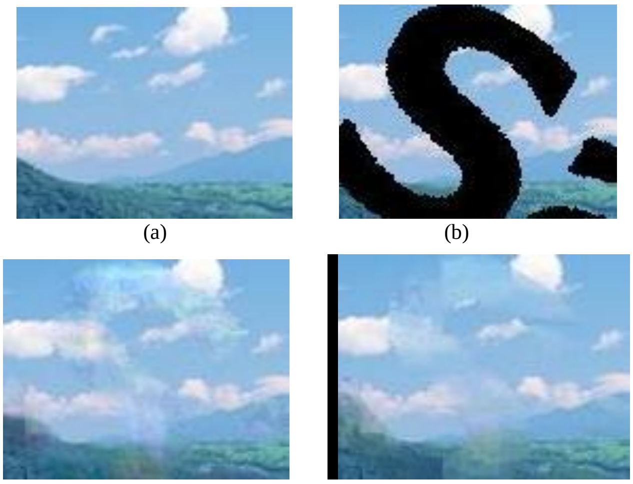
(d)

Fig. 7. The cropped part from the image and inpainting result: (a) original image (b) corrupted image (c) inpainting result by using original dictionary (d) inpainting result by using related dictionary.

In order to view the experiment results easily, Fig. 7 and Fig. 8 are cropped to show the inpainting result and the same parts of original image and corrupted image.

As shown in Fig. 5(c) and Fig. 6(c), there are some artifacts in the inpainting results by using the original dictionary, and they can be shown in Fig. 7(c) and Fig. 8(c) easily. These artifacts are produced because the original dictionary contains the unrelated atoms with the image patch. And the inpainting results by using related dictionary in Fig. 5(d) and Fig. 6(d) obtain good visual effect without a feeling of oddness, and they can also be shown in Fig. 7(d) and Fig. 8(d) easily.

From the subjective evaluation and objective evaluation, we can see clearly that the performance of inpainting algorithm using related dictionary are better than inpainting algorithm using original dictionary. This implies that the patch inpainting using related dictionary is more effective than using the original dictionary.



(a) (b) (c) (d)

Fig. 8. The cropped part from the image and inpainting result: (a) original image (b) corrupted image (c) inpainting result by using original dictionary (d) inpainting result by using related dictionary.

V. Conclusion

In this paper, we have proposed a new inpainting method based on sparse representation with related dictionaries. A related dictionary is well defined for each target patch, which consists of the patches in the uncorrupted area having a similar histogram with that of the patch. This is an important step, because it guarantees the inpainting result is more accurate. Thus, the target patch can be inpainted by a sparse representation of the patches in the related dictionary.

As shown in the experimental results, the proposed method using related dictionary has obtained better performance than using the original dictionary. The objective quantitative evaluation is consistent with the subjective visual effect of the inpainting result.

In the future, we will concern how to choose out the patches are more relational to the target patch by using the histogram. The chosen patches are more similar to the target patch, the inpainting results will be better. In addition, we also want to study other methods for image inpainting, especially the methods for the inpainting region is big.

References

[1] C. Guillemot and O. L. Meur, "Image inpainting: Overview and recent advances," *IEEE Signal Processing Magazine*, vol. 31, no. 1, pp. 127–144, Jan. 2014.

[2] M. J. Fadili, J.L. Starck, and F. Murtagh, "Inpainting and zooming using sparse representations," *The Computer Journal*, vol. 52, no. 1, pp. 64–79, Jan. 2009.

[3] T. Chan and J. Shen, "Local inpainting models and T inpainting," *SIAM J. Appl. Math*, vol. 62, no. 3, pp. 1019–1043.

[4] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image inpainting," in *Proc. the 27th Annual Conference on Computer Graphics and Interactive Techniques*, ACM Press/Addison-Wesley Publishing Co. ACM, New York, NY, 2000, pp. 417–424.

[5] A. A. Efros and T. K. Leung, "Texture synthesis by non-parametric sampling," in *Proc. the Seventh IEEE International Conference on Computer Vision*, Sept. 20-25, 1999, pp. 1033–1038.

[6] L. Liang, C. Liu, Y. Q. Xu, B. Guo, and H. Y. Shum, "Real-time texture synthesis by patch-based sampling," *ACM Transactions on Graphics (ToG)*, ACM, New York, NY, pp. 127-150, 2011.

[7] A. Criminisi, P. Pe ́rez, and K. Toyama, "Region filling and object removal by exemplar-based image inpainting," *IEEE Trans. on Image Processing*, IEEE Press Piscataway, NJ, pp. 1200–1212, Sept. 2004.

[8] A. Criminisi, P. Perez, and K. Toyama, " Object removal by exemplar-based inpainting," in *Proc. IEEE Computer Society Conference on Computer Vision & Pattern Recognition*, June 18-20, 2003.

[9] Z. Xu and J. Sun, "Image inpainting by patch propagation using patch sparsity," *IEEE Trans. on Image Processing*, vol. 19, no. 5, pp. 1153–1165, 2010.

[10] L. J. Deng, T. Z. Huang, and X. L. Zhao, "Exemplar-based image inpainting using a modified priority definition," *Plos One*, vol. 10, Oct. 2015.

[11] J. Mairal, M. Elad, and G. Sapiro, "Sparse representation for color image restoration," *IEEE Trans. on Image Processing*, vol. 17, no. 1, pp. 53–69, Jan. 2008.

[12] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Trans. on Image Processing*, vol. 15, no. 12, pp. 3736–3745, Dec. 2006.

[13] M. Aharon, M. Elad, and A. Bruckstein, "rmk-svd: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. on Signal Processing*, vol. 54, no. 11, pp. 4311–4322.

[14] B. Shen, W. Hu, Y. Zhang, and Y. J. Zhang, "Image inpainting via sparse representation," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Taipei, Taiwan, April 19-24, 2009, pp. 697–700.

[15] M. Elad, J. L. Starck, P. Querre, and D. L. Donoho, "Simultaneous cartoon and texture image inpainting using morphological component analysis (MCA)," *Applied and Computational Harmonic Analysis*, pp. 340–358.

[16] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Trans. on Information Theory*, vol. 53, no. 12, pp. 4655–4666, Dec. 2007.

[17] K. Xu, N. Wang, and X.Gao, "Image inpainting based on sparse representation with dictionary pre-clustering," in *Proc. Chinese Conference on Pattern Recognition*. Springer, pp. 245–258, 2016.

[18] M. Yaghoobi, D. Wu, and M. E. Davies, "Fast non-negative orthogonal matching pursuit," *IEEE Signal Processing Letters*, vol. 22, no. 9, pp. 1229–1233, Sept. 2015.

[19] Z. Wang and A. C. Bovik, "Mean squared error: Love it or leave it? A new look at signal fidelity measures," *IEEE Signal Processing Magazine*, vol. 26, no. 1, pp. 98–11, Jan. 2009.

Qiaoqiao Li is a Ph.D. student at Akita Prefectural University, Japan. She received the B.S. degree in electronic information engineering from Zhengzhou University, China, in 2012, and received the M.S. degree from Lanzhou University, China, 2016. Her main research is image inpainting.

G**uoyue Chen** received his BSc degree from East China Normal University, China, in 1983, where he worked as a research associate at the Department of Computer Science until 1989. He received his MS and PhD degrees from Tohoku University, Japan in 1993 and 1996, respectively. He is currently a professor in Akita Prefecture University, Japan. His interests include digital signal processing and its applications to active noise control, and image processing.

Xingguo Zhang received the B.E. degree in computer science from the Dalian Polytechnic University of China in 2009, and the M.E. and Ph.D. degrees from the Akita Prefectural University of Japan in 2012 and 2015. Currently, he is a teaching assistant in Akita Prefectural University, Japan. His research interests include computer vision, pattern recognition, and machine learning.

Kazuki Saruta received his BE and ME degrees from Akita University, Japan, in 1991 and 1993, and his PhD degree from Tohoku University, Japan, in 1996. Then, he was a lecturer in the Faculty of Humanities of Yamagata University. In 1999, he was invited as an associate professor in the Department of Electronics and Information Systems of Akita Prefectural University, Japan. He has been engaged in research on pattern recognition and image processing

Yuki Terata received his BE degree from Akita Prefecture University, Japan, in 1998, and the ME and PhD degrees from the same university in 2000 and 2008. He is a research associate in Akita Prefecture University, Japan. He is interested in VR/AR for cognitive psychology of elderly people.