# Image enlargement with high-frequency component augmentation based on predefined codebook describing edge blurring properties

OPTICAL REVIEW **Regular Paper**

# Image Enlargement with High-Frequency Component Augmentation Based on Predefined Codebook Describing Edge Blurring Properties

Hideaki Kawano*, Hakaru Tamukoh[1], Noriaki Suetake[2], Byungki Cha[3], and Takashi Aso[3]

*Faculty of Engineering, Kyushu Institute of Technology, 1-1, Sensui-cho, Tobata-ku, Kitakyushu 804-8550, Japan*

[1]*Faculty of Technology, Tokyo University of Agriculture and Technology, 2-24-16 Naka-chou, Koganei-shi, Tokyo 184-8588, Japan*

[2]*Graduate School of Science and Engineering, Yamaguchi University, 1677-1, Yoshida, Yamaguchi-shi, Yamaguchi 753-8512, Japan*

[3]*Faculty of Management and Information Sciences, Kyushu Institute of Information Sciences, 6-3-1, Saifu, Dazaifu-shi, Fukuoka 818-0117, Japan*

Image super-resolution as high-quality image enlargement is achieved by some type of restoration for high-frequency components that deteriorate through the image enlargement. The estimation methods using the given image itself are effective for the restoration, and we have proposed a method employing the codebook describing edge blurring properties that are derived from the given image. It is, however, unfavourable to apply those image-dependent methods to movies whose scene varies momentarily. In this paper, an image-independent codebook incorporating local edge patterns of images is proposed, and then the predefined codebook is applied. The effectiveness is shown through some experiments.

**Keywords:** image enlargement, super-resolution, codebook, edge pattern, image patch

## 1. Introduction

Advances in electronic imaging devices ranging from high-definition televisions to displays mounted on cell phones require more advanced image enlargement techniques. Classical image enlargement methods are based on interpolation using kernels, and include nearest-neighbor interpolation (NNI), bilinear interpolation (BLI), and cubic convolution interpolation (CCI).[1,2)] In the case in which the objective of the enlargement is only obtaining the smooth enlarged image, those interpolation-based methods are very useful and effective. However, they cannot be used to restore the high-frequency image components lost in the sampling process, and therefore cannot preserve clearly the step edges and peaks of the image. This is because high-frequency image components beyond the Nyquist frequency cannot be restored by these simple kernel-based methods. If multiple frames are available, a high-resolution image can be generated from a set of low-resolution frames in the same scene. Indeed, those methods[3–5)] produce a more similar result to the original image than interpolation methods

*E-mail address: kawano@ecs.kyutech.ac.jp

using a single frame, but they can not be applied to stationary images.

To address these problems, various advanced image enlargement methods with a single frame have been proposed.[6] For example, there are directional methods,[7,8] multiple-kernel-based methods,[9] orthogonal-transform-based methods,[10,11] linear extrapolation methods,[12] neural-network-based methods,[13–15] fuzzy-inference-based methods.[16,17] Although they have better performance than simple kernel-based methods, they are complex, making their practical implementation difficult. Furthermore, some of them are based on assumptions that are not always correct in practical situations, and thus they may give poor results.

Image super-resolutions as high-quality image enlargements are achieved by some type of restoration method for high-frequency components that deteriorate through the image enlargement. The estimation methods using the given image itself are effective for the restoration.[18–20] We have proposed an image enlargement method preserving perceptual sharpness,[19,20] which is achieved by augmenting a low-resolution enlarged image with high-frequency components estimated from a given image itself. The estimation of the high-frequency image components is performed using a codebook built by a decomposition of the given image, namely, a self-decomposed codebook. The rationale that is exploited in that approach is the shape-invariant properties of edges across scales. Indeed, the self-decomposed codebook could lead the enlargement to a satisfactory result, but the previous method suffers from high computational cost and a considerable memory storage requirement. The size of the self-decomposed codebook is approximately the number of pixels in the given image. Even worse, dependence on the given image makes it difficult for the previous method to expand to movies that change the scene momentarily.

To cope with those difficulties, a universal and compact codebook that can be usable in various scenes is proposed. We construct the compact predefined codebook considering edge blurring properties for various edge patterns in small image patches. Through our careful investigations, we derived the fundamental edge patterns. To show the effectiveness of the predefined codebook, we compared the enlarged results of the predefined codebook with those of the self-decomposed codebook. Consequently, we confirmed that the predefined codebook yields equivalent performance with respect to MSE evaluations.

## 2. Codebook-Based Image Enlargement Framework

In this section, the codebook-based image enlargement framework that has been proposed by the authors is briefly revisited.

### *2.1 Codebook generation*

In the framework, a codebook that describes image blurring properties is constructed from a given image. The term *image blurring property* means the relationship between a low-pass filtered (LPF) image $I_l$ and a high-pass filtered (HPF) image $I_h$ for the corresponding part

(a) (b) (c)

Fig. 1. Image blurring property in Lena. (a) A given image $I$, (b) LPF image $I_l$, (c) HPF image $I_h$.

of the given image $I$. Figure 1 shows an example of these images in *Lena*. Both $I_l$ and $I_h$ are decomposed to small, fixed-sized patches in a sliding manner. Let $x_l^p$ be a small image patch extracted from $I_l$ at arbitrary pixel coordinate $p$, and $x_h^p$ be a small image patch extracted from $I_h$ at arbitrary pixel coordinate $p$. An actual representation of the codebook is all pairs of patch pairs $\{x_l^p, x_h^p\}$ that are obtained from all over the image. The generated codebook is referred to as a *self-decomposed codebook* in this paper.

### *2.2 Codebook reference*

In the same way as we generated $x_l^p$ and $x_h^p$, a vector $X_l^q$ is generated with the same size patch from the cubic-convolution-interpolated image at arbitrary coordinate $q$. The vector $X_l^q$ is compared with all patches $x_l^p$, and the best-matching patch is determined on the basis of a local distance measure. Therefore, we seek a pixel coordinate $q$ for each $X_l^q$, which minimizes the following function:

$$p^* = \arg\min d(x_l^p, X_l^q), \tag{1}$$

where pixel coordinates $p$, $q$ run over all points in their respective images. The function $d(\cdot, \cdot)$ is a local distance measure between two small patches. In this paper, the Euclidean distance is used as below:

$$d(x_l^p, X_l^q) = ||x_l^p - X_l^q||. \tag{2}$$

### *2.3 Estimation of lost high-frequency components*

After the determination of the corresponding $x_h^{p^*}$ for all patches $X_l^q$ through the codebook reference mentioned above, all overlapped patches for a single pixel are dealt with as follows: In the case of a $4 \times 4$ patch size, the number of shared patches for a single pixel is 16 except for pixels near the edge of the image and in the corners. In such exceptional regions, a single pixel shares less patches. The corresponding pixel intensities in the shared patches are summed and then divided by the overlapped number. Consequently, the high-frequency components

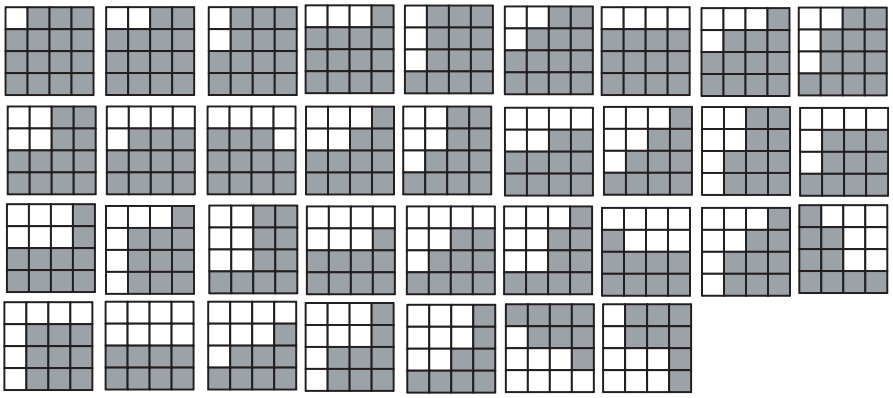


Fig. 2. Fundamental edge patterns.

of the enlarged image are created. The resultant high-frequency image is superimposed on the cubic-convolution-interpolated image to estimate the high-frequency components. Then we accomplish an image enlargement with perceptual sharpness.

## 3. Predefined Codebook

Indeed, the self-decomposed codebook could lead the enlargement to a satisfactory result, but the previous method suffers from high computational cost and a considerable memory storage requirement. The size of the self-decomposed codebook is approximately the number of pixels in the given image. Even worse, dependence on the given image makes it difficult for the previous method to expand to movies that change the scene momentarily.

As we can see in Fig. 1, high-frequency components to be augmented exist only on edge regions in a given image. In other words, flat regions would not contribute to the augmentation of high-frequency components. Therefore, we can only focus on pairs of small patches containing edge patterns. By carefully investigating pairs of small patches $\{x_l^p, x_h^p\}$ in various images, we contrived fundamental edge patterns, as shown in Fig. 2. These 34 small image patches represent typical variation of intensities in a $4 \times 4$ image patch. By considering rotations every 90 degrees, inversions, and 7 different intensity gaps on the basis of the fundamental edge patterns as shown in Fig. 3, we obtained a universal and compact predefined codebook involving $1,904$ code words.

In the case of image enlargement from a $128 \times 128$-size image to a $256 \times 256$-size image, the self-decomposed codebook involves $15,625$ code words. Therefore, the size of the predefined codebook could be reduced to approximately one-eighth of the self-decomposed codebook in this case. If the size of a given image is larger, the size of the self-decomposed codebook would become much larger. To realize the predefined codebook into an FPGA (Field Programmable Gate Array), it needs $548,352$ bits of FPGA internal memory. In particular, each codeword $\{x_l, x_h\}$ has 32 pixels, and these pixels are represented as 9 bits accuracy in the predefined

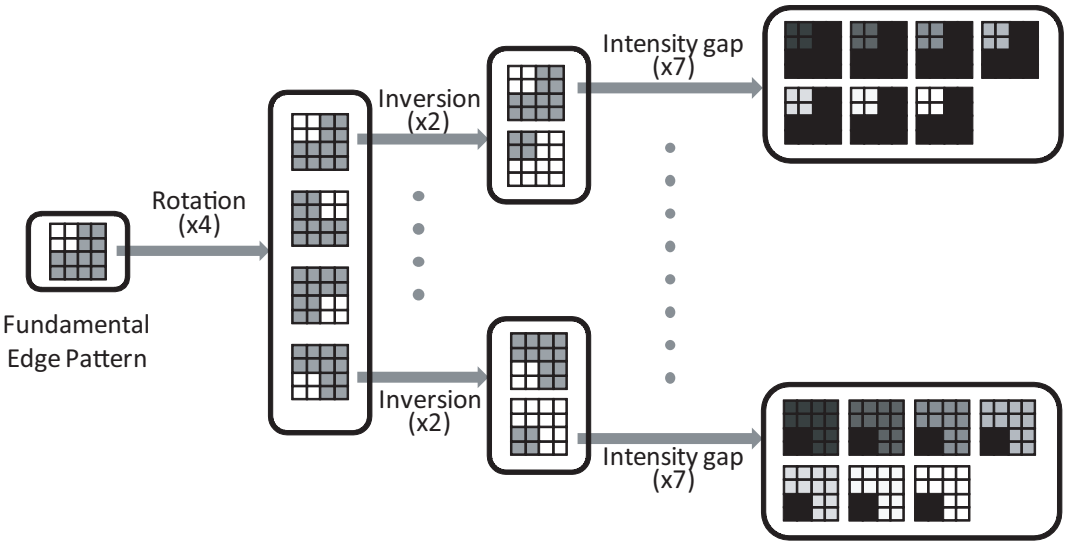



Fig. 3. Translated variants for a fundamental edge pattern.

codebook; thus, the total memory would be 548,352 bits [32 (pixel/word) × 9 (bit/pixel) × 1,904 (word)]. It is compact and easy to implement into the FPGA. For example, Xilinx xc3s4000, which is a low-cost FPGA built on CMOS 90 nm technology, has 1,769,472 bits Block RAM (FPGA internal memory).[21] Similarly, Xilinx xc6vsx475t, which is a recent high-performance FPGA built on CMOS 40 nm state-of-the-art technology, has 39,223,296 bits Block RAM.[21] Therefore, FPGA implementation of the proposed method is free from the memory limitation. Meanwhile, the process of eqs. (1) and (2) is commonly called as winner-take-all (WTA). Since WTA is widely used in the field of image processing and neural networks and so on, several digital hardware implementations have been proposed to realize a high-speed processing.[22,23] Furthermore, the proposed method has data parallelism similar to other image processing methods. For example, if the given image can be sliced into four divided images, four enlargement circuits can run in parallel. This means that the proposed method is suitable for massively parallel architecture and the hardware implementation would realize real-time processing. According to these estimates, the predefined codebook can be implemented on standard FPGA boards.

In the following, the procedure to obtain a $12 \times 12$ geometrically similar edge pattern from each $4 \times 4$ fundamental edge pattern is explained. The fundamental edge pattern is arranged at the center of a $12 \times 12$ pixel array, and then by preserving the arranged fundamental edge pattern as it is, the boundaries between the white region and the black region are extended in a natural way. Some examples of the geometrically similar edge patterns obtained by the above procedure are shown in Fig. 4. To generate such pairs of small patches as the self-decomposed codebook, we consider a $12 \times 12$ image patch that is geometrically similar to each $4 \times 4$ fundamental edge pattern. Then, LPF is applied to the $12 \times 12$ image patch. Finally,

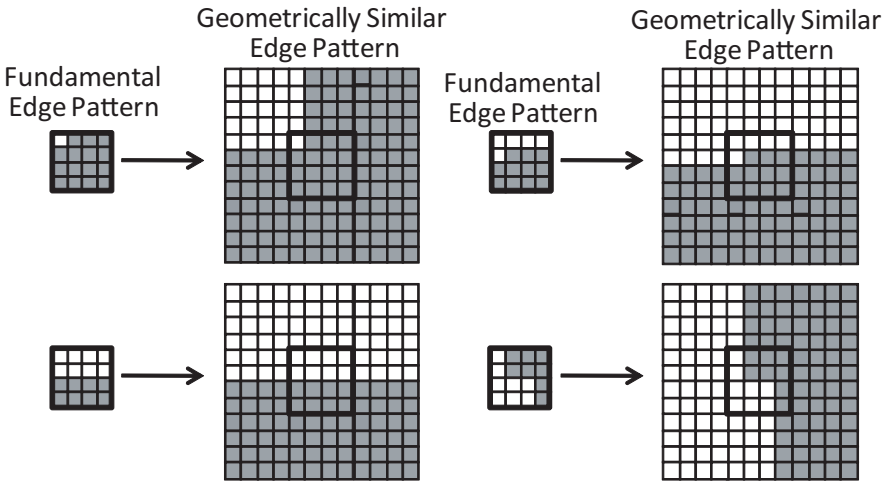



Fig. 4. Some examples of geometrically similar edge patterns.

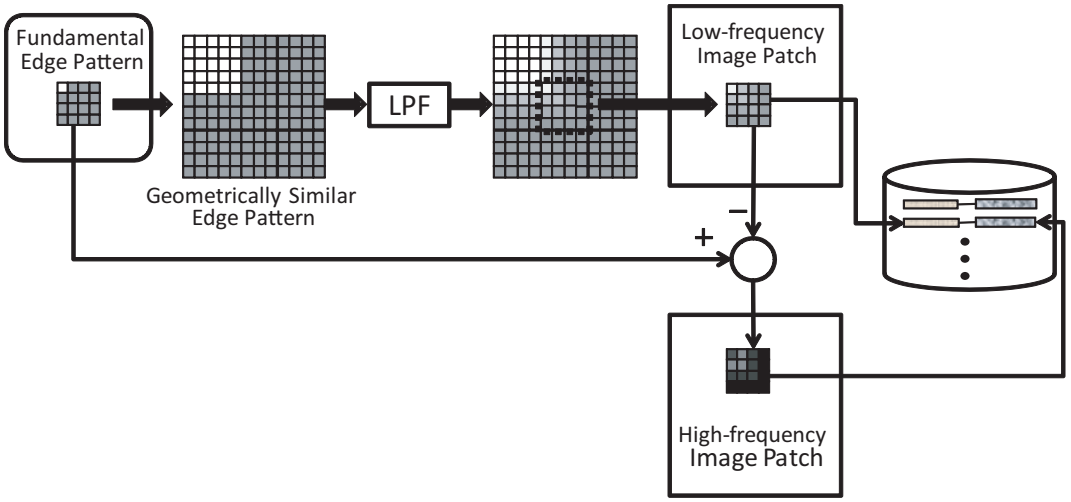



Fig. 5. Generation of predefined codebook.

a $4 \times 4$ image patch in the middle is extracted to obtain $x_l^p$. Meanwhile, the high-frequency components image patch $x_h^p$ is obtained by subtraction between the original patch and the LPF patch. For each fundamental edge pattern with various translations, a pair of low- and high-frequency image patches is stored as a codebook. These series of procedures with respect to the predefined codebook generation are summarized in Fig. 5.

## 4. Experimental Results

To show the effectiveness and validity of the predefined codebook, we compared the enlarged results of the predefined codebook with those of the self-decomposed codebook.

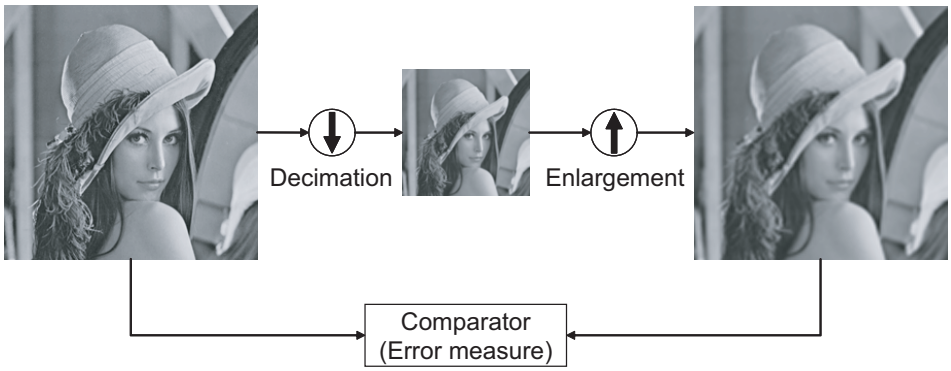



Fig. 6. Error measure layout for objective evaluation.

### *4.1 Setup*

Error measures are used to objectively compare the enlarged image with the original one, as shown in Fig. 6. The original image is first decimated by a factor of 2 and then enlarged by a factor of 2. The original and enlarged images are compared using an error measure. The decimation operation involves anti-aliasing filtering with the same parameter of LPF followed by down-sampling. As an error measure, the mean-squared error (MSE) is used in this paper. The MSE is simply the mean of the squared differences for every channel for every pixel. Letting $x_{ic}$ denote the value of pixel $i$ in channel $c$ of the original image, $y_{ic}$ the value of pixel $i$ in channel $c$ of the compared image, $n$ the number of pixels, and $c$ the number of channels, the MSE can be obtained using the following equation:

$$MSE = \frac{1}{nm} \sum_{i=1}^{n} \sum_{c=1}^{m} (x_{ic} - y_{ic})^2 . \tag{3}$$

The test set consists of the seven gray-scale images with an original resolution of $256 \times 256$ shown in Fig. 7. These images are royalty-free and have been used in other computer graphics performance tests, and are often referred to as standard images.

### *4.2 Results*

To perform objective evaluations, several enlargement algorithms are compared using the MSE measure. MSE results for enlargement are shown in Table 1 for gray-scale images. The best result in each row is marked in bold face. Three conventional methods were selected for comparison: cubic convolution interpolation (CCI),[2] nonlinear extrapolation (NE),[18] and the self-decomposed codebook-based enlargement (SDC).[20] The predefined codebook-based enlargement proposed in this paper (PDC) is compared with these methods. As shown in Table 1, the PDC enlargement method is on a par with the SDC enlargement method. Consequently, we confirmed that the predefined codebook yields equivalent performance with respect to MSE evaluations.

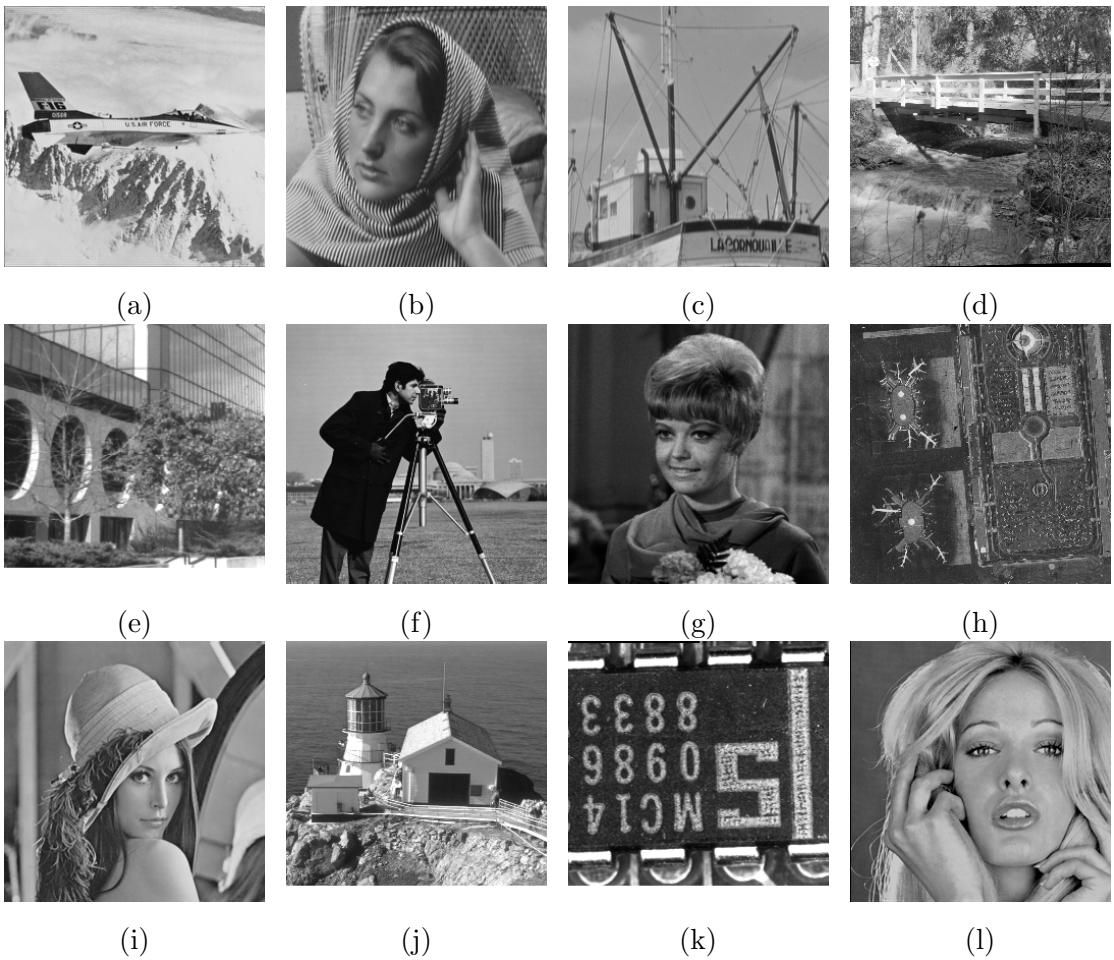


Fig. 7. Test set for gray-scale images: (a) Airplane, (b) Barbara, (c) Boat, (d) Bridge, (e) Building, (f) Cameraman, (g) Girl, (h) Lax, (i) Lena, (j) Lighthouse, (k) Text, and (l) Woman.

Table I. Results of MSE evaluations for various methods.

| | CCI | NE | SDC | PDC |
|---|---|---|---|---|
| Airplane | 123.8 | 118.5 | **91.0** | 101.6 |
| Barbara | 272.0 | 268.7 | **263.4** | 264.2 |
| Boat | 69.2 | 66.6 | **55.1** | 57.3 |
| Bridge | 299.8 | 287.2 | 284.2 | **274.2** |
| Building | 97.5 | 91.9 | **74.2** | 79.2 |
| Cameraman | 170.3 | 154.2 | **130.3** | 139.2 |
| Girl | 35.3 | 34.8 | **29.4** | 31.3 |
| Lax | 349.9 | 333.4 | **326.8** | 332.2 |
| Lena | 77.0 | 70.6 | **55.6** | 61.6 |
| Lighthouse | 272.1 | 255.7 | **236.8** | 237.1 |
| Text | 171.4 | 160.1 | **133.7** | 144.4 |
| Woman | 76.1 | 75.2 | **64.1** | 71.3 |

To assess the performance of the enlargement methods subjectively, we show the enlarged results with respect to "Lena" in Fig. 8. In Fig. 8, each image is a part of the original image cropped by $128 \times 128$. The images in Fig. 8 from left to right and top to bottom are: (a) ideal image, (b) the enlarged result of CCI, (c) the enlarged result of NE, (d) the enlarged result of SDC, and (e) the enlarged result of PDC. As shown in Fig. 8, the image by CCI is blurred. In contrast, the images by NE and our proposed methods SDC and PDC are comparatively sharper. We can see some artifacts along the shoulder line in the result of NE. In the comparison between SDC and PDC, we can observe that the result of SDC is smoother than the result of PDC in terms of the smoothness of the edge of the hat. This fact is backed by the numerical results in Table 1. The results of PDC tend to deteriorate in terms of the edge smoothness.

To provide a detailed description of the characteristics of each enlargement method, cross sections extracted from the ideal image and enlarged image by each method are shown in Fig. 9. A white line in Fig. 9(a) shows the location of the cross section. From the cross-sectional diagrams from Fig. 9(b) to 9(e), the following characteristics can be observed.

**CCI** The restoration of high-frequency components is still insufficient.

**NE** From the results of high-frequency components estimation, the edge is well preserved as compared with CCI. However, some degree of overshooting in the estimation can be observed. Thus, the deterioration such a ringing appeared in the image.

**SDC** In both the edge preservation and overshooting problem, the SDC exhibits superior characteristics. The deterioration such a ringing does not occur in the image enlarged by the SDC. We can observe that the SDC inhibits a ringing in comparison with NE.

**PDC** In both the edge preservation and overshooting problem, the PDC is comparable to the SDC despite a much considerably lower number of codebooks.

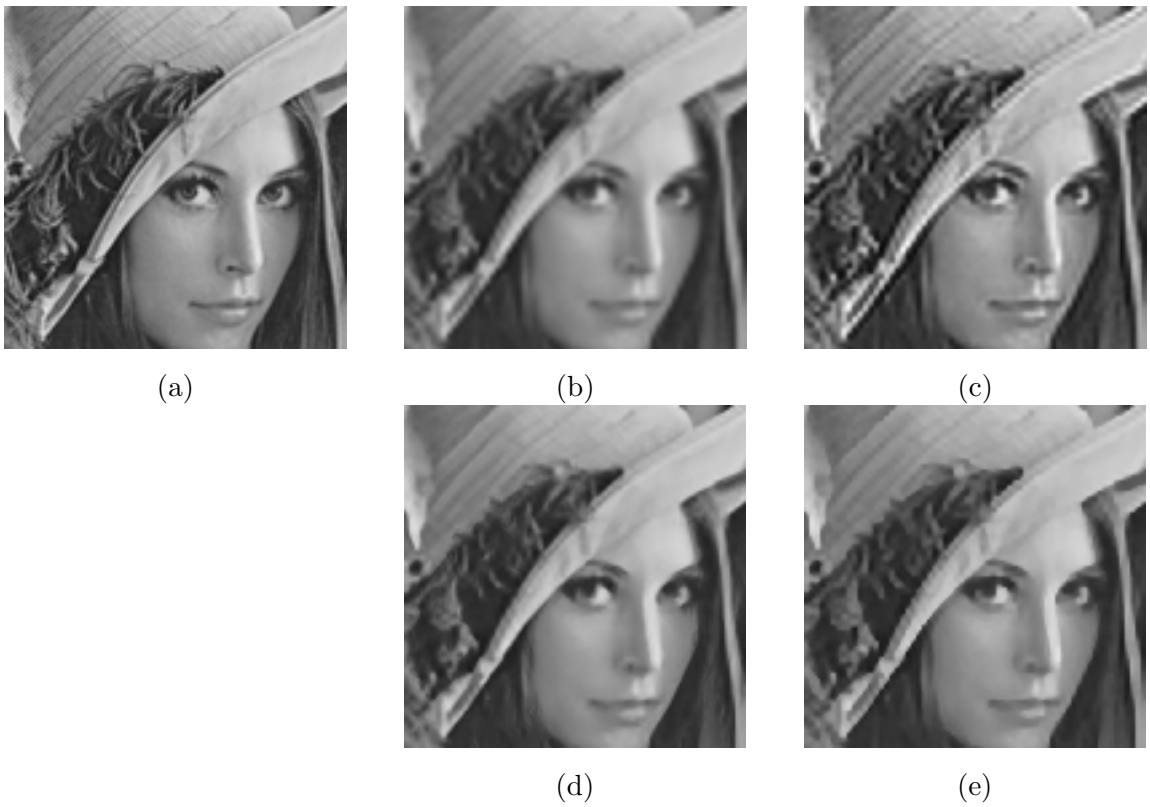
From these observations, the codebook-based image enlargement works well and the predefined codebook produces performance comparable to the self-decomposed codebook.

## 5. Conclusions

In this paper, to deal with the difficulties in the self-decomposed codebook-based image enlargement, we proposed the predefined codebook describing edge blurring properties.

From the experiments, it has been confirmed that the results also show that the performance of the proposed method is superior to that of other conventional image enlargement methods.

The algorithm of the proposed method is less complex and hence should be easily implementable in hardware. Additionally, the proposed enlargement method can be processed in parallel because the estimations of the high-frequency image patches for every LPF image patch are calculated independently. Therefore, the proposed method has potential for a wide



(a) (b) (c)

(d) (e)

Fig. 8. Original gray-scale image and the enlargement results: (a) Ideal, (b) CCI, (c) NE, (d) SDC, and (e) PDC.

range of image enlargement applications. In future work, the proposed image enlargement based on a predefined codebook will be implemented on hardware and applied to movies.

**Acknowledgment**

This work was supported by the Japan Society for the Promotion of Science under a Grant-in-Aid for Scientific Research C (No. 21500230).

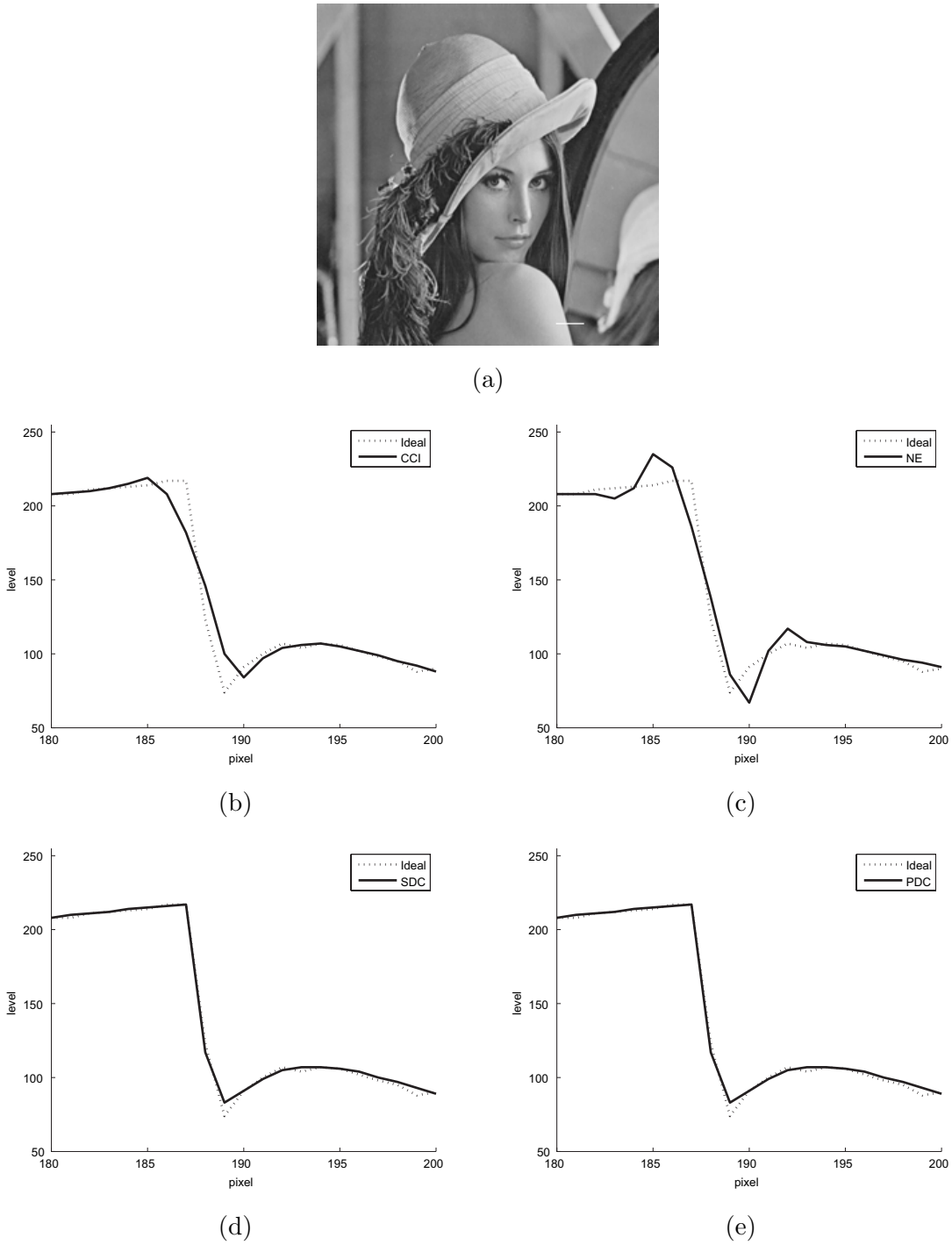



Fig. 9. Cross sections with respect to ideal image and the result of each method: (a) Location, (b) CCI, (c) NE, (d) SDC, and (e) PDC.

**References**


1) J. S. Lin: *Two-Dimensional Signal Processing and Image Processing* (Prentice-Hall, Inc., New Jersey, 1990) p. 495.

2) R. G. Keys: IEEE Trans. Acoust. Speech Signal Process., **29** (1981) 1153.

3) S. C. Park, M. K. Park, and M. G. Kang: IEEE Signal Process. Mag., **20** (2003) 21.

4) M. Irani and S. Peleg, J. Vis. Commun. Image Represent., **4** (1993) 324.

5) Z. Lin and H. -Y. Shum: IEEE Trans. Pattern Anal. Mach. Intell. **26** (2004) 83.

6) J. D. van Ouwerkerk: Image Vision Compt., **24** (2006) 1039.

7) K. Jensen, and D. Anastassiou: IEEE Trans. Image Process. **4** (1995) 285.

8) X. Li, and M. T. Orchard: IEEE Trans. Image Process. **10** (2001) 1521.

9) A. M. Darwish, and M. S. Bedair: Proc. SPIE, **2466** (1996) 131.

10) S. A. Martucci: Proc. IEEE Int. Conf. Image Process., 1995, p. 2244.

11) S. G. Chang, Z. Cvetkovic, and M. Vetterli: Proc. IEEE Int. Conf. Acoust. Speech Signal Process., **4**, 1995, p. 2379.

12) T. Aso, N. Suetake, T. Yamakawa: Intelligent Automation and Soft Computing, **12** (2006) 345.

13) F. M. Candocia, and J. C. Principe: IEEE Trans. Neural Netw., **10** (1999) 372.

14) N. Plaziac: IEEE Trans. Image Process., **8** (1999) 1647.

15) F. Pan, and L. Zhang: Opt. Eng. **42** (2003) 3038.

16) A. Taguchi and T. Kimura: Proc. SPIE, **4304** (2001) 98.

17) S. Carrato, G. Ramponi, and S. Marsi: Proc. IEEE Int. Conf. Image Process., 1996, p. 711.

18) H. Greenspan, C. H. Anderson, and S. Akber: IEEE Trans. Image Process., **9** (2000) 1035.

19) N. Suetake, M. Sakano, and E. Uchino: Opt. Rev., **15** (2008) 26.

20) H. Kawano, N. Suetake, B. Cha, and T. Aso: Image Vision Comput., **27** (2009) 684.

21) Xilinx Inc.: http://www.xilinx.com/, Ref. Mar. 1, 2010.

22) D. C. Hendry: Neurocomputing, **62** (2004) 389.

23) H. Tamukoh, K. Horio, and T. Yamakawa: IEICE Trans. Electron., **E87-C** (2004) 1787.