Band-Pass $\varepsilon$-Filter for Edge Enhancement and Noise Removal

Mitsuharu MATSUMOTO$^a$, Member

SUMMARY A band-pass bilateral filter is an improved variant of a bilateral filter that does not have low-pass characteristics but has band-pass characteristics. Unfortunately, its computation time is relatively large since all pixels are subjected to Gaussian calculation. To solve this problem, we pay attention to a nonlinear filter called $\varepsilon$-filter and propose an advanced $\varepsilon$-filter labeled band-pass $\varepsilon$-filter. As $\varepsilon$-filter has low-pass characteristics due to spatial filtering, it does not enhance the image contrast. On the other hand, band-pass $\varepsilon$-filter does not have low-pass characteristics but has band-pass characteristics to enhance the image contrast around edges unlike $\varepsilon$-filter. The filter works not only as a noise reduction filter but also as an edge detection filter depending on the filter setting. Due to its simple design, the calculation cost is relatively small compared to the band-pass bilateral filter. To show the effectiveness of the proposed method, we report the results of some comparison experiments on the filter characteristics and computational cost.

key words: noise reduction, edge enhancement, nonlinear filter, band-pass $\varepsilon$-filter

1. Introduction

Filtering plays an important role in image processing and computer vision. In terms of noise removal, low-pass filtering such as linear filter and Gaussian filter works well in smooth regions, but significantly blurs the edge structures of an image. To solve this problem, although there are many studies on reducing noise while preserving the edge [1]–[7], the bilateral filter is an attractive filtering method that preserves the edge information [8]. It employs two types of Gaussian filter executed in the spatial - range domain. The spatial low pass Gaussian filter gives higher weights to pixels that are spatially close to the center pixel. The range low pass Gaussian filter gives higher weights to pixels that are similar to the center pixel in gray value. Combining the range filter and the domain filter, a bilateral filter at an edge pixel becomes an elongated Gaussian filter that is oriented along the edge, which ensures that averaging is done mostly along the edge and is greatly reduced in the gradient direction. It has various applications such as image processing [9] and mesh denoising [10].

Although it can reduce noise while preserving the edge, as bilateral filter also has low-pass characteristics due to the spatial filtering, it does not enhance the image contrast. As an improved bilateral filter, Inoue et al. has proposed a band-pass bilateral filter (BP bilateral filter) [11] which does not have low-pass characteristics but has band-pass characteristics to enhance edges. However, it requires a lot of computation time because of the Gaussian calculation in all pixels.

To solve the problem, we look to a nonlinear filter labeled $\varepsilon$-filter [12]–[14]. $\varepsilon$-filter is a nonlinear filter, which can reduce noise while preserving the edge like the bilateral filter. The algorithm is simple and the calculation cost is small compared to the bilateral filter. However, $\varepsilon$-filter still has low-pass characteristics and therefore does not enhance the image contrast around edges.

Hence, we propose band-pass $\varepsilon$-filter (BP $\varepsilon$-filter). The band-pass characteristics are formulated by employing the concept of Difference of Gaussian (DoG). DoG is an image enhancement filter which is based on the difference between different low-pass filters with Gaussian window. In spite of its simple design, DoG has many applications such as edge detection, image enhancement and automatic scale selection. By combining the concept of $\varepsilon$-filter and DoG, the calculation cost of BP $\varepsilon$-filter can be relatively small like $\varepsilon$-filter. The filter works not only as a noise reduction filter but also as an edge detection filter depending on the filter setting.

In the next section, we first describe the algorithm of the conventional $\varepsilon$-filter. In Sec.3, we describe the algorithm of the BP $\varepsilon$-filter. In Sec.4, we describe the experimental evaluation of its filter characteristics. We compare the calculation cost with some other filterings. The robustness of edge detection from noisy data is also evaluated. Discussions and conclusion follow in Sec.5.

2. $\varepsilon$-Filter

We firstly explain the algorithm of $\varepsilon$-filter to clarify its features. For ease of understanding, we first describe the one-dimensional case. Let us define $x(k)$ as the input signal (For instance, the signal including speech signal with noise) at time $k$. Let us also define $y(k)$ as the output signal of $\varepsilon$-filter at time $k$ as follows:

$$
y(k) = \Phi_{\varepsilon,M}[x(k)] = x(k) + \sum_{i=-M}^{M} a(i) F(x(k + i) - x(k)),
$$

where $a(i)$ represents the filter coefficient. $a(i)$ is usually constrained as follows:

Copyright © 2010 The Institute of Electronics, Information and Communication Engineers
all the points where the distance from \( \varepsilon \) as 

\( F(k) \)

ample, we can set the nonlinear function

\[ F(x) = \begin{cases} 
  x & (-\varepsilon \leq x \leq \varepsilon) \\
  0 & (\text{else}) 
\end{cases} \] (4)

Figure 1 shows the basic concept of \( \varepsilon \)-filter when we utilize Eq. 4 as \( F(x) \). Figure 1 (a) shows the waveform of the input signal. Executing \( \varepsilon \)-filter at point A in Fig. 1 (a), we replace all the points where the distance from A is more than \( \varepsilon \) by the value of point A. We then summate the signals in the same window. Figure 1 (b) shows the basic concept of this procedure. In Fig. 1 (b), the dotted line represents the points where the distance from A is more than \( \varepsilon \). In Fig. 1 (b), the solid line represents the values replaced through this procedure. As a result, if the points are far from A, the points are ignored. On the other hand, if the points are close to A, the points are smoothed. Because of this procedure, \( \varepsilon \)-filter reduces the noise while preserving the precipitous attack and decay of the speech signal. In the same way, executing \( \varepsilon \)-filter at point B in Fig. 1 (a), we replace all the points where the distance from B is more than \( \varepsilon \) by the value of point B. The points are ignored if they are far from B, while the points are smoothed if the points are close to B. Consequently, we can reduce the small amplitude noise near the processed point while preserving the speech signal.

\( \varepsilon \)-filter can easily be improved not only for one dimension but also for two dimensions. Let us define \( I(k, l) \) as the two dimensional image data at \((k, l)\). Let us define \( J(k, l) \) as the output of a two dimensional \( \varepsilon \)-filter. When we apply \( \varepsilon \)-filter to two dimensional data such as an image, \( \varepsilon \)-filter is designed as follows:

\[ J(k, l) = \Phi_{\varepsilon, M}[I(k, l)] = I(k, l) + \sum_{i=-M}^{M} \sum_{j=-M}^{M} A(i, j) F(I(k + i, l + j) - I(k, l)), \] (5)

where \( A(i, j) \) represents the filter coefficient. \( A(i, j) \) is usually constrained as follows:

\[ \sum_{i=-M}^{M} \sum_{j=-M}^{M} A(i, j) = 1. \] (6)

The feature of a two dimensional \( \varepsilon \)-filter is similar to that of a one dimensional \( \varepsilon \)-filter. We can smooth the small amplitude noise near the processed point while preserving the edge. It requires less calculation when it is compared to a bilateral filter because it requires only switching and linear operation. However, due to this procedure, \( \varepsilon \)-filter basically contains low-pass characteristics.

3. Band-Pass \( \varepsilon \)-Filter

Band-pass \( \varepsilon \)-filter is an improved \( \varepsilon \)-filter which does not have low-pass characteristics but has band-pass characteristics. Band-pass characteristic is based on the idea of difference of Gaussian (DoG). DoG is a band-pass filter, which involves the subtraction of one blurred version of an original grayscale image from another less blurred version of the original. The blurred images are obtained by convolving the original grayscale image with Gaussian kernels having differing standard deviations. Blurring an image using Gaussian kernels suppresses only high-frequency spatial information. Subtracting one image from the other preserves spatial information that lies between the range of frequencies that are preserved in the two blurred images. Due to these reasons, DoG is equivalent to a band-pass filter. It is designed as follows:

\[ J(k, l) = \Psi_{\varepsilon, 1}[I(k, l)] - \gamma \Psi_{\varepsilon, 2}[I(k, l)], \] (7)
where
\[
\Psi_{r}[I(k, l)] = \sum_{i=-M}^{M} \sum_{j=-M}^{M} \exp\left(\frac{-(k+i)^2-(l+j)^2}{2\sigma^2}\right) I(k+i, l+j). \tag{9}
\]

\(\gamma\) represents the filter coefficient and is constrained as follows:
\[
0 \leq \gamma \leq 1. \tag{10}
\]

\(\sigma_1\) and \(\sigma_2\) are constrained as follows:
\[
0 < \sigma_1 < \sigma_2. \tag{11}
\]

In band-pass \(\varepsilon\)-filter, we employ \(\varepsilon\)-filter instead of Gaussian filter. The band-pass \(\varepsilon\)-filter (BP \(\varepsilon\)-filter) is designed as follows:
\[
J(k, l) = \Phi_{\varepsilon_1, M_1}[I(k, l)] - \gamma \Phi_{\varepsilon_2, M_2}[I(k, l)] \tag{12}
\]

In \(\varepsilon\)-filter, when \(\varepsilon\) and \(M\) are large, the image is strongly smoothed, while the image is less blurred when \(\varepsilon\) and \(M\) are small. Hence, \(\varepsilon_1\) and \(\varepsilon_2\) are constrained as follows:
\[
0 < \varepsilon_1 \leq \varepsilon_2. \tag{13}
\]

\(M_1\) and \(M_2\) are also constrained as follows:
\[
0 < M_1 \leq M_2. \tag{14}
\]

\(\gamma\) is the filter coefficient constrained as follows:
\[
0 \leq \gamma \leq 1. \tag{15}
\]

By changing \(\gamma\) values from 0 to 1, BP \(\varepsilon\)-filter changes its characteristic from \(\varepsilon\)-filter-like to DoG-like. In BP \(\varepsilon\)-filter, \(\varepsilon_1\) and \(\varepsilon_2\) work as the parameters concerning signal intensity, while \(M_1\) and \(M_2\) work as the spatial parameters.

4. Experiment

4.1 Filter Characteristics with Changing \(\gamma\) Values

To evaluate the filter characteristics of BP \(\varepsilon\)-filter, we firstly show the experimental results with changing \(\gamma\) values. As an example, we show the image using “Lena” as shown in Fig. 2 (a). Figure 2 (b) shows the image with white noise. To clarify how noise is added, we cut and enlarged the image around the face. Figure 3 shows the output of BP \(\varepsilon\)-filter when \(\gamma\) value was set to 0. In this case, BP \(\varepsilon\)-filter works as a simple \(\varepsilon\)-filter. It reduced the noise while preserving the edge as shown in Fig. 3. Figures 4 (a), 4 (b), 4 (c) and 4 (d) show the output of BP \(\varepsilon\)-filter when \(\gamma\) values were set to 0.25, 0.5, 0.75, and 1, respectively. Throughout the experiments, \(\varepsilon_1\) and \(\varepsilon_2\) were set to 30 and 50, respectively. \(M_1\) and \(M_2\) were set to 1. As shown in Figs. 3, 4 (a), 4 (b), 4 (c) and 4 (d), the BP \(\varepsilon\)-filter has both characteristics of \(\varepsilon\)-filter and DoG. When \(\gamma\) was close to 1, it worked as an edge detection filter similar to DoG. In this regard, however, it should be noted that the noise was reduced while preserving the edge due to
the difference not of two Gaussian filters but of two $\varepsilon$-filters. To show the difference between DoG and the BP $\varepsilon$-filter, we also show the output images of DoG as shown in Fig. 5 with changing $\gamma$. As shown in Fig. 5, although DoG reduced image noise, it also blurred the edge of the image when $\gamma$ is equal to 0.25, 0.5 and 0.75. On the other hand, BP $\varepsilon$-filter can reduce noise while preserving the edge of the image. By setting the $\gamma$ value between 0 and 1, we can enhance the image contrast around the edge because BP $\varepsilon$-filter acts like a mix of $\varepsilon$-filter and DoG.

4.2 Filter Characteristics with Changing $\varepsilon$ Values

It is considered that $\varepsilon$ and $M$ determine what kind of edges BP $\varepsilon$-filter enhances. For instance, when we set both $M_1$ and $M_2$ to the same value, the two $\varepsilon$-filters equally smooth or ignore the image regions whose variations are less than $\varepsilon_1$ or more than $\varepsilon_2$. Hence, BP $\varepsilon$-filter will enhance the edge of the image regions whose variation is between $\varepsilon_1$ and $\varepsilon_2$. In other words, band-pass $\varepsilon$-filter enhances edge, whose intensity gradient is constrained as:

$$\varepsilon_1 < \sqrt{\left(\frac{\partial I(k, l)}{\partial k}\right)^2 + \left(\frac{\partial I(k, l)}{\partial l}\right)^2} < \varepsilon_2.$$  \hspace{1cm} (16)

To clarify the above points, we also conducted experiments on evaluating filter characteristics with changing $\varepsilon$ values. The window size of both $\varepsilon$-filters is set to 3[pixels] $\times$ 3[pixels]. For the benchmark analysis, we prepared a test image which included six lines with different intensity gradient as shown in Fig. 6. The value of the intensity gradient of the lines are also described in Fig. 6. As shown in Fig. 6, the intensity gradient increases from the upper line (intensity

Fig. 3  Output image of BP $\varepsilon$-filter when $\gamma = 0$ (Simple $\varepsilon$-filter).

Fig. 4  Output image of BP $\varepsilon$-filter with changing $\gamma$ values.

Fig. 5  Output image of Difference of Gaussian with changing $\gamma$ values.

Fig. 6  Test image with six lines of different intensity gradient.
gradient: 40) to the lower line (intensity gradient: 240) with interval of 40. Figure 7 shows the experimental results using the test image. In the experiments, we set six pairs of $\varepsilon_1$ and $\varepsilon_2$ as follows:

$$ (\varepsilon_1, \varepsilon_2) = (20, 60), (60, 100), (100, 140), (140, 180), (180, 220), (220, 260), \quad (17) $$

As shown in Fig. 7, BP $\varepsilon$-filter enhances the line which has an intensity gradient between $\varepsilon_1$ and $\varepsilon_2$. We can utilize its features depending on the application. For instance, we can enhance weak edge while reducing noise from the highly damaged image.
4.3 Filter Characteristics with Changing \( M \) Values

When \( \varepsilon_1 \) and \( \varepsilon_2 \) are set to the same \( \varepsilon \) value, two \( \varepsilon \)-filters equally ignore the regions whose variations are more than \( \varepsilon \). Hence, BP \( \varepsilon \)-filter will enhance the image edge regions whose variations are less than \( \varepsilon \). In other words, BP \( \varepsilon \)-filter enhances edges, whose intensity gradient is constrained as:

\[
\sqrt{(\frac{\partial I(k,l)}{\partial k})^2 + (\frac{\partial I(k,l)}{\partial l})^2} < \varepsilon.
\]  

(18)

We conducted the experiments on evaluating the filter characteristics with changing \( M \) values. In the experiments, we set \( M_1 \) and \( M_2 \) to 3 and 5, respectively. We set six pairs of \( \varepsilon_1 \) and \( \varepsilon_2 \) as follows:

\[
(\varepsilon_1, \varepsilon_2) = (40, 40), (80, 80), (120, 120),
(160, 160), (200, 200), (240, 240),
\]

(19)

As shown in Fig. 8, BP \( \varepsilon \)-filter enhances the line which has an intensity gradient of less than \( \varepsilon_1 \) and \( \varepsilon_2 \).

To clarify the general features of the BP \( \varepsilon \)-filter, we show other examples of BP \( \varepsilon \)-filter. We employed several images listed in standard image database (SIDBA) and show three examples of them. Figures 9, 10 and 11 show the results when we used “Barbara”, “Boat”, and “Cameraman”, respectively. The upper two images in Figs 9, 10 and 11 are the original image and the image with noise, respectively. The lower left images in Figs 9, 10 and 11 are the outputs of BP \( \varepsilon \)-filter when we set the \( \gamma \) values of BP \( \varepsilon \)-filter to 0, that is, it works as the simple \( \varepsilon \)-filter. The lower right images in Figs 9, 10 and 11 are the outputs of BP \( \varepsilon \)-filter when we set the \( \gamma \) values of BP \( \varepsilon \)-filter to 1, that is, it works as an edge detector. As shown in Figs 9, 10 and 11, the BP \( \varepsilon \)-filter works not only as a noise reduction filter but also as an edge detection filter depending on the parameter settings in each of the cases.

4.4 Evaluation of Calculation Cost

We also conducted an experiment to confirm that the calculation cost is smaller than other methods such as difference of Gaussian, bilateral filter and band-pass bilateral filter. We used a computer with an Intel Core 2 Duo 1.58 GHz CPU. The program was implemented by MATLAB. Figure 12 shows the calculation cost depending on the window size concerning the band-pass \( \varepsilon \)-filter and other methods such as DoG, bilateral filter and BP bilateral filter. In the experiments, the window sizes of BP \( \varepsilon \)-filter were set to the same values to compare it with other methods under the hardest condition although it could be smaller because the window size of one \( \varepsilon \)-filter can be set to a small value compared to that of the other \( \varepsilon \)-filter. The window size changed from \( 3[\text{pixels}] \times 3[\text{pixels}] \) to \( 9[\text{pixels}] \times 9[\text{pixels}] \). We used three images to confirm the calculation cost. The image size was \( 256[\text{pixels}] \times 256[\text{pixels}] \). As shown in Fig. 12, the calculation cost of the proposed method was small compared not
only to BP bilateral filter but also to the simple DoG and bilateral filter.

4.5 Evaluation of Edge Detection Performance

We also conducted experiments to evaluate the robustness of band pass \( \varepsilon \)-filter with regard to edge detection from a noisy image. To evaluate edge detection performance quantitatively, we used figure of merit (FOM) defined as follows [15]:

\[
F = \frac{1}{\max(I_I, I_A)} \sum_{i=1}^{I_A} \frac{1}{1 + \alpha d^2(i)}
\]

(20)

where \( I_I \) and \( I_A \) are the number of ideal and detected edge points, respectively. \( d(i) \) is the distance between the true edge and the \( i \)th detected edge. \( \alpha \) is a scaling constant chosen to be \( \alpha = 1/9 \). High FOM value represents high performance for edge detection. FOM value is equal to 1 when the obtained edge is wholly correct.

We prepared a test image for edge detection as shown in Fig. 13. The image size is 256 [pixels] \( \times \) 256 [pixels]. We added various levels of random noise with uniform distribution to the original image. The maximal intensity of noise changes from 0 to 120 with interval of 5. When the maximal intensity of noise is \( J \), the noise range is \([−J, J] \).

Figure 14 shows the relation between noise intensity and FOM value. For comparison, we show FOM values not only of band-pass \( \varepsilon \)-filter, but also of Laplacian filter, Sobel filter \((g_{HS})\), Sobel filter \((g_{VS})\), Prewitt filter \((g_{HS})\), Prewitt filter \((g_{VS})\) and DoG. \( g_{HS} \) and \( g_{VS} \) represent the filter for detecting horizontal gradient and vertical gradient, respectively.

As shown in Fig. 14, band-pass \( \varepsilon \)-filter maintains high performance in spite of noise corruption even when the noise intensity is 120, while FOMs of the other methods become worse when the image is highly corrupted. As the test image only includes the horizontal gradient, Sobel filter \((g_{vx})\) and Prewitt filter \((g_{vx})\) do not detect anything throughout the experiments. As noise is randomly added, detection error occurs randomly. FOM becomes better if the error occurs near true edge, while FOM becomes worse if it occurs far from true edge. Hence, it is considered that the vibration of FOM is due to the randomness of the noise.
To clarify the performance of band-pass $\varepsilon$-filter, we also show the filter outputs when the noise intensity is 120. Figure 15 shows the test image corrupted with noise. Figure 16 (a) shows true edge obtained from the original test image. Figures 16 (b), 16 (c), 16 (d), 16 (e) and 16 (f) show the filter outputs of Laplacian filter, Prewitt filter, Sobel filter, DOG and band-pass $\varepsilon$-filter, respectively. As shown in Figs. 16 (b), 16 (c), 16 (d) and 16 (e), the filter outputs of the typical edge detection filters include not only the true edge but also noise information. However, band-pass $\varepsilon$-filter can adequately detect true edge without detecting noise information.

5. Discussions and Conclusion

In this paper, we proposed a band-pass $\varepsilon$-filter (BP $\varepsilon$-filter), which works not only as a noise reduction filter while preserving an edge information but also as edge detection filter. It has both characteristics of $\varepsilon$-filter and DoG. It could reduce noise while preserving the edge when $\gamma$ was set close to 0, and it could detect the edge while reducing the noise when $\gamma$ was set close to 1. If $\gamma$ was set close to 0, it becomes a simple $\varepsilon$-filter. Hence, the proposed method corresponds to the special case of an $\varepsilon$-separating nonlinear filter bank in the reference [14]. The calculation cost was relatively small compared to conventional filters such as not only BP bilateral filter but also the simple DoG and bilateral filter. According to the experimental results, it is possible to use BP $\varepsilon$-filter not only for noise reduction but also for edge detection. $\varepsilon$ and $M$ determine what kind of edges BP $\varepsilon$-filter enhances. The quantitative characteristic features about parameter changes are also evaluated theoretically and empirically. There are simple relationships among the intensity gradient, the $\varepsilon_1$ and $\varepsilon_2$. This clear relation among the intensity gradient, the $\varepsilon_1$ and $\varepsilon_2$, will help the user to set parameter. Future works are to develop the optimal setting of each $\varepsilon$ and $M$ depending on the application and to apply the proposed method to medical image analysis including those for cancer and tumors to assist in the making of medical diagnostics.

Acknowledgement

This research was supported by Special Coordination Funds for Promoting Science and Technology, by a research grant of Support Center for Advanced Telecommunications Technology Research (SCAT), by a research grant of Foundation for the Fusion of Science and Technology, by a research grant of Tateisi Science and Technology Foundation, and by the Ministry of Education, Science, Sports and Culture, Grant-in-Aid for Young Scientists (B), 20700168, 2008.

References

[1] A. Bovik, T. Huang, and D. Munson, “Image restoration using order-constrained least-squares methods,” Proc. IEEE International Conference on Acoustics Speech, and Signal Processing, pp.828–831, 1983.

[2] T. Boult, R. A. Melter, F. Skorina, and I. Stojmenovic, “G-neighbors,” Proc. SPIE Conference on Vision Geometry II, SPIE,
vol.2060, pp.96–109, 1993.

[3] R.T. Chin and C.L. Yeh, “Quantitative evaluation of some edge preserving noise-smoothing techniques,” CVGIP, vol.23, pp.67–91, 1983.

[4] L.S. Davis and A. Rosenfeld, “Noise cleaning by iterated local averaging,” IEEE Trans. Syst. Man Cybern., vol.SMC-8, no.9, pp.705–710, 1978.

[5] N. Himayat and S.A. Kassam, “Approximate performance analysis of edge preserving filters,” IEEE Trans. Signal Process., vol.41, no.9, pp.2764–2777, 1993.

[6] T.S. Huang, G.J. Yang, and G.Y. Tang, “A fast two-dimensional median filtering algorithm,” IEEE Trans. Acoust., Speech Signal Process., vol.27, no.1, pp.13–18, 1979.

[7] L. Yin, R. Yang, M. Gabbouj, and Y. Neuvo, “Weighted median filters: A tutorial,” IEEE Trans. Circuits Syst., vol.43, no.3, pp.157–192, 1996.

[8] C. Tomasi and R. Manduchi, “Bilateral filtering for gray and color images,” Proc. International Conference on Computer Vision, pp.839–846, 1998.

[9] R. Ramanath and W.E. Snyder, “Demosaicking as a bilateral filtering process,” Proc. SPIE, SPIE, vol.4667, no.5, pp.236–244, 2002.

[10] S. Fleishman, I. Drori, and D. Cohen-Or, “Bilateral mesh denoising,” Proc. ACM SIGGRAPH, vol.4667, no.5, pp.950–953, 2003.

[11] K. Inoue, W. Du, and K. Urahama, “Image smoothing with edge enhancement by using band-pass bilateral filter,” IEICE Trans. Fundamentals (Japanese Edition), vol.J88-A, no.5, pp.687–690, May 2005.

[12] H. Harashima, K. Odajima, Y. Shishikui, and H. Miyakawa, “ε-separating nonlinear digital filter and its applications,” IECE Trans. (Japanese Edition), vol.J65, no.4, pp.297–304, April 1982.

[13] K. Arakawa, K. Matsuura, H. Watabe, and Y. Arakawa, “A method of noise reduction for speech signals using component separating ε-filters,” IEICE Trans. Fundamentals (Japanese Edition), vol.J85-A, no.10, pp.1059–1069, Oct. 2002.

[14] K. Arakawa and T. Okada, “ε-separating nonlinear filter bank and its application to face image beautification,” IEICE Trans. Fundamentals (Japanese Edition), vol.J88-A, no.11, pp.1216–1225, Nov. 2005.

[15] I.E. Abdou and W.K. Pratt, “Quantitative design and evaluation of enhancement/thresholding edge detectors,” Proc. IEEE, vol.67, no.5, pp.753–763, 1979.

Mitsuharu Matsumoto was born in 1978. He received a B.S. in Applied Physics, and M.E. and Dr. Eng. in Pure and Applied Physics from Waseda University, Tokyo, Japan, in 2001, 2003, and 2006, respectively. He has published 13 books and more than 50 papers in refereed internal conferences and journals. He was a research associate and an assistant professor in Waseda University from 2006 to 2008. He is currently an assistant professor in the University of Electro-Communications. His research interests include array signal processing, blind source separation, image processing, optical device, pattern recognition, self-assembly and robotics. He received Ericsson Young Scientist Award from Nippon Ericsson K.K., Japan, in 2009. He is a member of the Institute of Electrical and Electronic Engineers (IEEE).