An FFT-Based Block Matching Algorithm Using Asymmetric Search Window

Hidetake SASAKI and Hitoshi KIYA
Department of Information and Communication Systems, Tokyo Metropolitan University
6–6 Asahigaoka, Hino-shi, Tokyo 191–0065, Japan
E-mail: sasaki-hidetake@sd.tmu.ac.jp, kiya@sd.tmu.ac.jp

***Abstract*—This paper proposes a novel algorithm for weighted sum of squared differences (SSD)-based block matching algorithm (BMA) and its application to nonlocal-means (NLM) filtering. The proposed method is based on the BMA using the fast Fourier transformation and employs an asymmetric search window, then the proposed method allows us to simultaneously compute two weighted SSDs in the NLM. The method further reduces the number of FFT operations by sharing a common search window among multiple pixels. Experimental results show that the proposed method is about 1.8 or more times faster than the conventional spatial domain-based NLM filtering.**

I. Introduction

Block matching is widely used in many fields, such as pattern recognition, inpainting, and image denoising. Therefore, many fast algorithms have been proposed before now. There are two main types of these algorithms. One type operates in the spacial domain. The diamond search [1] and the successive elimination search [2] are representatives of this type.

The other type of fast block matching algorithms (BMAs) is based on the fast Fourier transformation (FFT) [3]–[6]. Compared to the direct full-search BMA which exhaustively searches for every possible candidate in the search window to find the most similar block, this type of BMAs greatly decrease the computational load without the complete loss of accuracy. Most conventional methods of this type calculate a sum of squared differences (SSD), thus some applications such as nonlocal-means (NLM) filter [7] which needs to calculate weighted SSDs rather than SSDs can not employ the FFT-based BMA.

The NLM filter is a high-quality denoising filter. The method is essentially a neighborhood filter, in which a noisy pixel-value is replaced by a weighted average of the pixel-values in the entire noisy image, where the weights are determined by the neighborhood similarity between a pixel of interest (POI) and another pixel. Thanks to its patch-based approach, images denoised by the NLM filtering maintain fine structure and details. However, the disadvantage of the NLM filter is a high computational cost for its pixel-wise window matching.

FFT-based methods are proposed [8], [9] to overcome the above-mentioned problem. These methods, however, reduce the accuracy of a denoised image owing to the use of SSD criterion for calculation of neighborhood similarity between a POI and another pixel.

This paper proposes a novel FFT-based BMA with weighted SSD criterion. The proposed method employs an asymmetric search window (ASW) which utilizes the characteristics of FFT-based calculation. By employing the proposed method, computational cost of the NLM is greatly decreased with the same accuracy as the conventional spatial domain-based method [7].

II. Preliminary

A. Weighted SSD Cost Function

For search window $f(x,y)$ of size $M \times N$ and block $b(x,y)$ of size $A \times B$, the SSD which is generally used as a criterion in FFT-based BMAs can be expressed as

$$\mathrm{SSD}_{b,f}(u,v) = \sum_{x=0}^{A-1}\sum_{y=0}^{B-1} \{f(x+u,y+v)-b(x,y)\}^2, \quad (1)$$

$$u \in [0, M-A+1], \quad v \in [0, N-B+1], \quad (u,v) \in \mathbb{Z}^2,$$

where variables u and v are shift amounts.

On the other hand, the weighted SSD is computed as follows,

$$\mathrm{WSSD}_{w,b,f}(u,v) = \sum_{x=0}^{A-1}\sum_{y=0}^{B-1} w(x,y)\{f(x+u,y+v)-b(x,y)\}^2, \quad (2)$$

$$u \in [0, M-A+1], \quad v \in [0, N-B+1], \quad (u,v) \in \mathbb{Z}^2,$$

where $w(x,y)$ is a weighting function over $b(x,y)$. The weighted SSD is equivalent to SSD when all elements of $w(x,y)$ are 1.

B. Nonlocal-Means Filter

NLM [7] is an image denoising filter, in which a noisy pixel-value is replaced by a weighted average of the pixel-values in the entire noisy image, where the weights are determined by the weighted SSD between a POI and another pixel.

In the NLM, given noisy image $f(x,y)$ defined on $\Omega \subset \mathbb{R}^2$, estimated value $\mathrm{NL}_f(x_i,y_i)$ for a POI at (x_i,y_i) is derived as

$$\mathrm{NL}_f(x_i,y_i) = \frac{1}{C(x_i,y_i)} \sum_{(x_j,y_j)\in\Omega} d_i(x_j,y_j) f(x_j,y_j), \quad (3)$$

where weight $d_i(x_j,y_j)$ of two pixels at (x_i,y_i) and (x_j,y_j) depends on their neighborhood similarity $\mathrm{WSSD}_{w_a,f_i,f_j}(x_j,y_j)$

and $C(x_i, y_i)$ is a normalizing constant. They are defined as,

$$d_i(x_j, y_j) = \exp\left(-\frac{\mathrm{WSSD}_{w_a, f_i, f_j}(x_j, y_j)}{h^2}\right), \quad (4)$$

$$C(x_i, y_i) = \sum_{(x_j, y_j) \in \Omega} d_i(x_j, y_j), \quad (5)$$

where h acts as a filtering degree and $\mathrm{WSSD}_{w_a, f_i, f_j}(x_j, y_j)$ is computed as the weighted SSD between two pixels' neighborhoods with equal size $M \times M$ as,

$$\mathrm{WSSD}_{w_a, f_i, f_j}(x_j, y_j) = \sum_{x=0}^{M-1} \sum_{y=0}^{M-1} w_a(x, y) \{ f(x+x_j, y+y_j) - f(x+x_i, y+y_i) \}^2, \quad (6)$$

where $w_a(x,y)$ is a two-dimensional Gaussian kernel with standard deviation a.

The measurement of the neighborhood similarity is the most costly part in the NLM filtering. In an effort of high-speed computing the costly part, this paper proposes an FFT-based calculation method with keeping the accuracy.

III. Proposed Method

A. Calculate a Weighted SSD Using FFT

The proposed method extends the conventional FFT-based BMA [5] to compute the weighted SSD. With the extended FFT-based BMA, computational cost of the neighborhood similarity is greatly decreased with the same accuracy as the conventional NLM filtering [7].

Firstly, Eq. (6) is expanded as Eq. (1) is expanded in the conventional FFT-based BMA,

$$\mathrm{WSSD}_{w_a, f_i, f_j}(x_j, y_j) = C_i - 2\mathrm{cor}_{w_a f_i, f_j}(x_j, y_j) + \mathrm{cor}_{w_a, f^2}(x_i, y_i), \quad (7)$$

where

$$C_i = \sum_{x=0}^{M-1} \sum_{y=0}^{M-1} w_a(x, y) f^2(x+x_i, y+y_i), \quad (8)$$

$$\mathrm{cor}_{w_a f_i, f_j}(x_j, y_j) = \sum_{x=0}^{M-1} \sum_{y=0}^{M-1} w_a(x, y) f(x+x_i, y+y_i) f(x+x_j, y+y_j), \quad (9)$$

$$\mathrm{cor}_{w_a, f^2}(x_j, y_j) = \sum_{x=0}^{M-1} \sum_{y=0}^{M-1} w_a(x, y) f^2(x+x_j, y+y_j). \quad (10)$$

C_i, the first term on the right side in Eq. (7), is constant because it is independent from (x_j, y_j). $\mathrm{cor}_{w_a f_i, f}(x_j, y_j)$ is a cross-correlation which can be easily calculated by FFT as,

$$\mathrm{cor}_{w_a f_i, f}(x_j, y_j) = F^{-1}\{F^*\{w_a(x,y) f(x+x_i, y+y_i)\} F\{f(x+x_j, y+y_j)\}\}, \quad (11)$$

where F and F^{-1} denote the FFT and the inverse FFT, respectively. The asterisk denotes complex conjugation. Equation (11) shows that the cross-correlation of two signals can be evaluated by multiplying corresponding Fourier coefficients of the two signals, followed by an inverse FFT of the result. $\mathrm{cor}_{w_a, f^2}(x_j, y_j)$, a cross-correlation between $w_a(x,y)$ and $f^2(x+x_i, y+y_i)$, can be calculated in a similar way. Figure 1 is a block diagram of the extended FFT-based BMA.

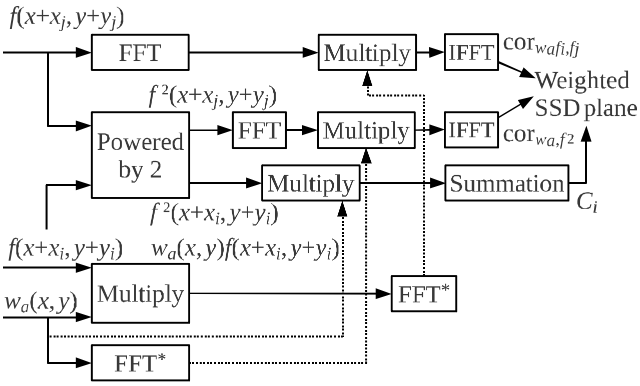
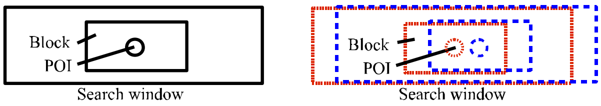


Fig. 1. Block diagram for extended FFT-based BMA.



(a) SSW. (b) SSW when the POI is shifted.

Fig. 2. Conceptual diagram for SSW (conventional [7]).

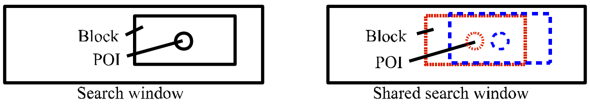
For an image with $n \times n$ pixels, $O(n^4)$ flops are taken for direct calculation of cross-correlation. On the other hand, the FFT-based calculation of cross-correlation takes only $O(n^2 \log n)$ flops. Therefore, the extended FFT-based BMA greatly decreases the computational cost of the neighborhood similarity calculation with the same accuracy as the conventional spatial domain-based method [7].

B. Asymmetric Search Window

Furthermore, by taking advantage of the FFT which is designed for complex signals, FFT-based methods can be more accelerated. If two blocks share the same search window, two weighted SSDs can be calculated at the same time by combining two real signals into a complex signal before the FFT is applied to.

In the conventional NLM [7], a POI is always centered in a search window as shown in Fig. 2 (a), which is called as a symmetric search window (SSW) hereafter. The SSW shifts with the POI as shown in Fig. 2 (b), thus a different search window is selected for each POI. So, the above-mentioned simultaneous calculation using complex-formed two real signals can not be applied because search windows for two POIs are different.

In contrast, an ASW employed in the proposed method need not locate the POI at the center as shown in Fig. 3 (a). Thus the ASW can be shared for two or more successive POIs as shown in Fig. 3 (b). In the proposed method, therefore, the ASW and the extended FFT-based BMA allow us to simultaneously compute two weighted SSDs in the NLM. Furthermore, the ASW decreases the number of FFTs because of the sharing search window for several POIs more than two.



(a) ASW. (b) ASW when the POI is shifted.

Fig. 3. Conceptual diagram for ASW (proposed).

IV. SIMULATIONS

In this section, the effectiveness of the proposed method is evaluated by implementing the NLM filtering. Experiments were implemented in the 32 bits version MATLAB and performed on a computer with an Intel Core i7 870 CPU and 7.8GiB RAM.

A. Conditions

White Gaussian noise with different standard deviation σ are added to 8-bit grayscale test images, and then the proposed method is compared to the conventional method in terms of run-time and the peak signal-to-noise ratio (PSNR) between the original image and the denoised image. Four methods were compared in our experiments: conventional method [7] which computes weighted SSDs in the spatial domain, "SSW" which calculates those by FFT-based BMA with SSW, "ASW4" which calculates those by FFT-based BMA with ASW sharing each search window for four blocks, and "ASW36" which calculates those by FFT-based BMA with ASW sharing each search window for 36 blocks.

The conditions are listed in Table I. Three test images were used in our experiments: Penny (128×128 pixels), Aerial (256×256 pixels), and Ruler (300×300 pixels). Standard deviations σ of white Gaussian noise which were added to test images were 10, 20, and 20, respectively. The filtering parameter h has been fixed to σ. Block size is fixed to 5×5 because it has shown to be large enough for being robust to noise and simultaneously small enough for taking care of details and fine structure. search window sizes for test images were 21×21, 21×21, and 65×65, respectively.

TABLE I
SIMULATION CONDITIONS

	Image size	Search window size	Block size	σ	h
Penny	128×128	21×21	5×5	10	10
Aerial	256×256	21×21	5×5	20	20
Ruler	300×300	65×65	5×5	20	20

TABLE II
RUN-TIME AND PSNR

Image		Conv. [7]	SSW	ASW4 (proposed)	ASW36 (proposed)
Penny	Run-time [sec]	37.9	5.3	2.8	2.6
	PSNR [dB]	35.1	35.1	35.1	35.0
Aerial	Run-time [sec]	156.3	19.7	11.2	10.5
	PSNR [dB]	25.0	25.0	25.0	25.0
Ruler	Run-time [sec]	1937.2	243.4	127.1	108.6
	PSNR [dB]	31.2	31.2	31.2	31.2

B. Results

Table II shows run-time and the PSNR of four different NLMs. As can be seen from Table II, image distortion derived from asymmetricity between the search window and the block is vanishingly small and the FFT-based methods ("SSW", "ASW4", and "ASW36") are about 8 times or more faster than the conventional spatial domain-based method [7]. Furthermore, by employing the ASW, the proposed method is about 14 times faster than the conventional spatial domain-based method [7]. "ASW36" is slightly faster than "ASW4". This acceleration is due to reduction in FFT operations by ASW described in Sec. III-B.

Figures 4 to 6 shows the result images of the experiments. Results of the method "SSW" are omitted here because results of "SSW" are exactly the same as those of the conventional method. As can be seen from these figures, differences between results of the proposed method and those of the conventional method which are derived from asymmetricity between the search window and the block are imperceptible.

V. CONCLUSIONS

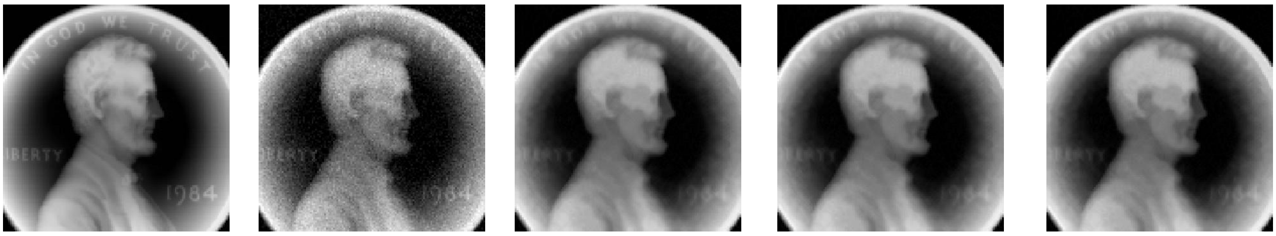
This paper proposes a novel algorithm for weighted SSD-based BMA with ASW and its application to NLM filtering. Two or more blocks can share the same search window by the ASW, thus the proposed method allows us to simultaneously compute two weighted SSDs in the NLM, and the method further reduces the number of FFTs in the calculation of the weighted SSD. Experimental results show that the proposed method is about 14 or more times faster than the non-FFT-based BMA in NLM [7] without significant loss of accuracy. It is 1.8 or more times faster even compared to the FFT-based BMA with SSW.

ACKNOWLEDGMENT

This work has been partly supported by the Grant-in-Aid for Scientific Research (C), No.20560361, from Japan Society for the Promotion of Science.

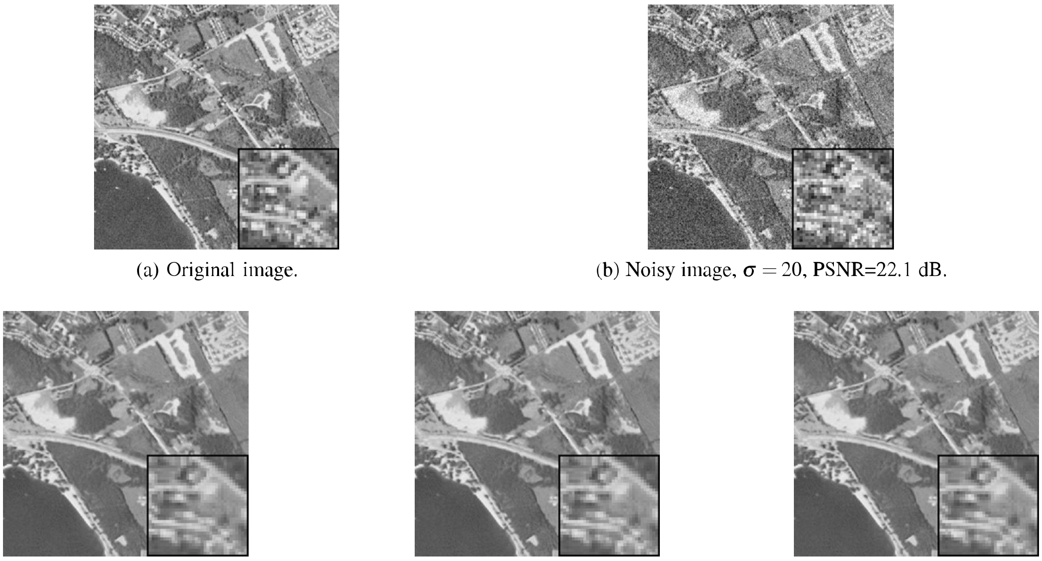
REFERENCES

[1] S. Zhu and K. Ma, "A new diamond serch algorithm for fast block motion estimation," *IEEE Trans. Image Process.*, vol.9, no. 2, pp.287–290, 2000.
[2] W. Li and E. Salari, "Successive elimination algorithm for motion estimation," *IEEE Trans. Image Process.*, vol.4, no.1, pp.105–107, 1995.
[3] S.L. Kilthau, M.S. Drew, and T. Moller, "Full search content independent block matching based on the fast fourier transform," in *Proc. IEEE ICIP*, 2002, pp.669–672.
[4] F. Essannouni, R.O.H. Thami, D. Aboutajdine, and A. Salam, "Simple noncircular correlation method for exhaustive sum square difference matching," *Opt. Eng.*, vol.46, pp.107004-1–107004-4, 2007.
[5] Z. Li and H. Kiya,"An FFT-based full-search block-matching algorithm with sum of squared difference criterion," *IEICE Trans. Fundamentals*, vol.E93-A, pp.1748–1754, Oct. 2010.
[6] H. Sasaki, Z. Li, and H. Kiya,"FFT-based full-search block matching using overlap-add method," in *Proc. PCS*, 2010, no.P4-28, pp.586–589.
[7] A. Buades, B. Coll, and J.-M. Morel, "A non-local algorithm for image denoising," in *Proc. IEEE CVPR*, vol.2, 2005, pp.60–65.
[8] J. Wang, Y. Guo, Y. Ying, Y. Liu, and Q. Peng, "Fast non-local algorithm for image denoising," in *Proc. IEEE ICIP*, 2006, pp.1429–1432.
[9] Y.-L. Liu, J. Wang, X. Chen, Y.-W. Guo, and Q.-S. Peng, "A robust and non-local means algorithm for image denoising," *J. Computer Science and Technology*, vol.23, no.2, pp.270–279, Mar. 2008.



(a) Original image. (b) Noisy image, $\sigma = 10$, PSNR=28.1 dB. (c) Conventional [7], PSNR=35.1 dB. (d) ASW4, PSNR=35.1 dB. (e) ASW36, PSNR=35.0 dB.

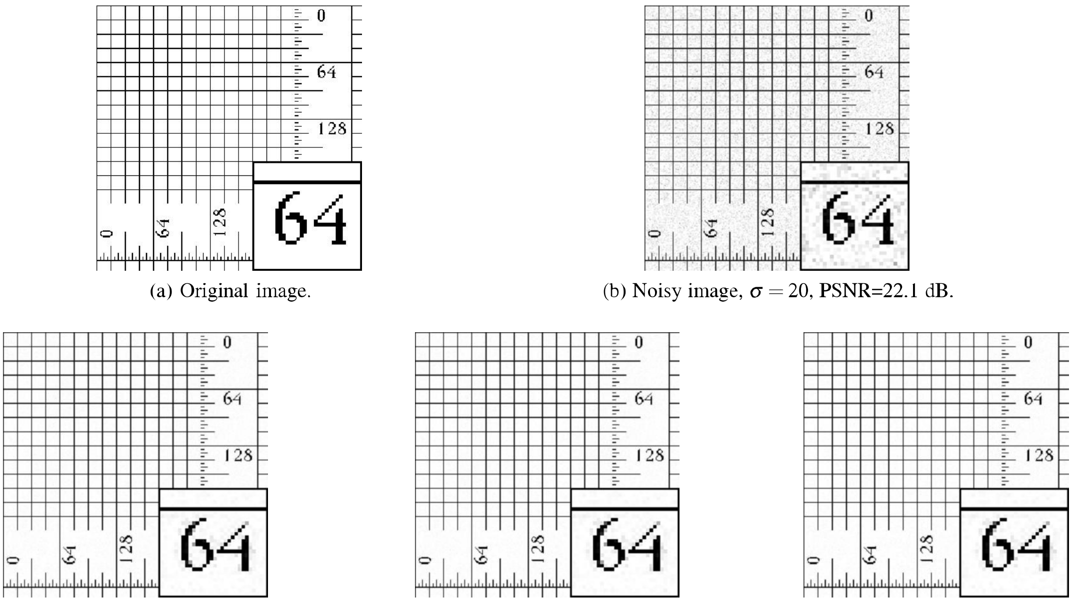
Fig. 4. Resulting images (Penny, 128×128 pixels).



(a) Original image. (b) Noisy image, $\sigma = 20$, PSNR=22.1 dB.

(c) Conventional [7], PSNR=25.0 dB. (d) ASW4, PSNR=25.0 dB. (e) ASW36, PSNR=25.0 dB.

Fig. 5. Resulting images (Aerial, 256×256 pixels).



(a) Original image. (b) Noisy image, $\sigma = 20$, PSNR=22.1 dB.

(c) Conventional [7], PSNR=31.2 dB. (d) ASW4, PSNR=31.2 dB. (e) Asw36, PSNR=31.2 dB.

Fig. 6. Resulting images (Ruler, 300×300 pixels).