SWITCHING DUAL KERNELS FOR SEPARABLE EDGE-PRESERVING FILTERING

Norishige Fukushima, Shu Fujita and Yutaka Ishibashi

Nagoya Institute of Technology, Japan

ABSTRACT

In this paper, we propose an accurate approximation framework for separable edge-preserving filtering. Naïve implementation of edge-preserving filtering, such as bilateral filtering and non-local means filtering, consumes enormous computational costs. Separable implementation of such filters is an efficient approximation method for real-time filtering. The accuracy of the conventional separable representation, however, is inadequate when the kernel size is immense. To improve the accuracy, we prepare dual kernels that have different kernel weights for horizontal and vertical filtering of separable filtering. In the experiment, we validate the proposed implementation by using three kinds of filters; bilateral filtering, dual bilateral filtering, and non-local means filtering. Experimental results show that the proposed implementation has higher accuracy while the computational time is almost the same. Moreover, the proposed implementation is practical for denoising and disparity map refinement applications.

***Index Terms*—** Edge-preserving filter, Separable filter, Bilateral filter, Real-time processing, Computational photography

1. INTRODUCTION

Edge-preserving filters are important tools for image processing, computer vision researchers. The representative filtering is the bilateral filter [1]. The edge-preserving filters are used for various applications, including image denoising [2], high dynamic range imaging [3], detail enhancement [4, 5], flash/no-flash photography [6, 7], up-sampling/super resolution [8], alpha matting [9, 10], haze removing [11], and optical flow or stereo correspondence problem [12, 13, 14, 15], its refinement processing [16, 17], coding noise removing [18, 19], and free viewpoint image rendering [20].

For real-time applications, efficient implementation or approximation is essential. Accordingly, tremendous number of acceleration methods are proposed [9, 21, 22, 23, 24, 25, 26, 27, 28]. Contrary to what we might think, the seminal acceleration approach of the separable implementation [29, 30] is the fastest approach yet within the practical kernel size.

The approximation approach of the separable implementation forcedly decomposes a filtering kernel into horizontal and vertical strips. The computational order of a naïvely implemented filter is $O(r^2)$, where r is a kernel radius of the filter. In constant, that of a separably implemented filter is $O(r)$. Even if the state-of-the-art filters have $O(1)$ order, i.e. the filters are independent of these kernel radius, the separable implementation defeats them as the aspect of the computational cost up to the middle kernel radius. Moreover, the straightforward implementation is suitable to build in circuits; therefore, the separable filtering is adequate for real-time applications. The issue of this approximation is its accuracy. It is not sufficient, and streaking noises are outstanding.

To improve the accuracy, we proposed novel implementation for the separable edge-preserving filtering. The proposed implementation prepares a new kernel for filtering at the separable second-pass. Moreover, we extend the separable implementation to applying various edge-preserving filters. We named the framework switching dual kernels based separable filtering (*SDK-SF*). The implemented codes and additional results are avalable from our website[1].

2. RELATED WORK

General edge-preserving filtering of finite impulse response (FIR) filtering is represented as:

$$\bar{\boldsymbol{I}}_{\boldsymbol{p}} = \frac{1}{K_{\boldsymbol{p}}} \sum_{\boldsymbol{q}\in\Omega} w_{\boldsymbol{p},\boldsymbol{q}} \boldsymbol{I}_{\boldsymbol{q}}, \tag{1}$$

where $\boldsymbol{p}, \boldsymbol{q}$ are center and reference pixel positions, $\boldsymbol{I_q}$ is a pixel value of an input image at $\boldsymbol{q}$, $\bar{\boldsymbol{I}}_{\boldsymbol{p}}$ is one of an filtered image at $\boldsymbol{p}$, Ω is a set in a kernel, $w_{\boldsymbol{p},\boldsymbol{q}}$ is a weight between $\boldsymbol{p}$ and $\boldsymbol{q}$, and $K_{\boldsymbol{p}} = \sum_{\boldsymbol{q}\in\Omega} w_{\boldsymbol{p},\boldsymbol{q}}$ is a normalized factor.

The weight of the edge-preserving filtering is defined by various types of filtering. It has domain and range kernel parts. The weight of the bilateral filtering [1] is denoted as:

$$w^{bi}_{\boldsymbol{p},\boldsymbol{q}} = \exp(\frac{-(\|\boldsymbol{p}-\boldsymbol{q}\|_2^2}{2\sigma_s^2}) \exp(\frac{-\|\boldsymbol{I_p}-\boldsymbol{I_q}\|_2^2}{2\sigma_r^2}), \tag{2}$$

where $\|\cdot\|_2$ is L2 norm, σ_s and σ_c are standard deviations for domain and range kernels. The weight of the dual bilateral filtering [31, 18, 12, 32] (also called trilateral / multilateral filtering) is defined as:

$$w^{dual}_{\boldsymbol{p},\boldsymbol{q}} = w^{bi}_{\boldsymbol{p},\boldsymbol{q}} \exp(\frac{-\|\boldsymbol{J_p}-\boldsymbol{J_q}\|_2^2}{2\sigma_{ar}^2}), \tag{3}$$

[1] http://fukushima.web.nitech.ac.jp/research/separable.html

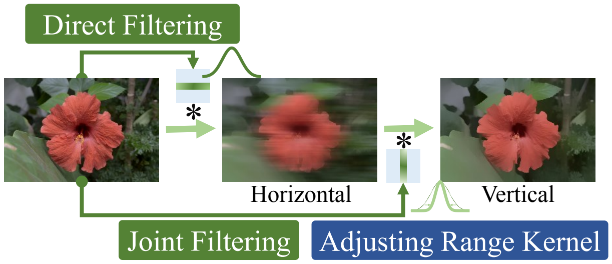

Fig. 1: Processing flow of switching dual kernels based separable filtering (SDK-SF).

where $\boldsymbol{J}$ is an additional joint image and σ_{ar} is a standard deviation for an extra range kernel. Thus, the filter has dual range kernels. The weight of the non-local means filtering [2] is defined as:

$$w^{nlm}_{\boldsymbol{p},\boldsymbol{q}} = \exp\left(\frac{-\|\boldsymbol{v}(\mathcal{N}_{\boldsymbol{p}}) - \boldsymbol{v}(\mathcal{N}_{\boldsymbol{q}})\|_2^2}{h^2}\right), \tag{4}$$

where h is a parameter for smoothing strength, and $\boldsymbol{v}(\mathcal{N}_{\boldsymbol{p}}) = \{I_{\boldsymbol{p}_1}, I_{\boldsymbol{p}_2}, \cdots, I_{\boldsymbol{p}_N}\}$ is a vectorizing function of neighborhood pixels. The function converts a set $\mathcal{N}_{\boldsymbol{p}}$, which is N neighborhood pixels around $\boldsymbol{p}$, to a $N \times 1$ vector $\boldsymbol{v}$.

Separable filtering is one of the efficient technique. Arbitrary kernel can be represented as a summation of separable kernels by using singular value decomposition (SVD) [33]. The SVD based method is effective for a spatially invariant kernel. In the case of spatially variant kernel, the conventional approaches decompose kernels directly for separable bilateral filtering [29] and separable non-local means filtering [30] . With the implementation, an input image is filtered only by the horizontal direction at the first pass, and then, the horizontally filtered image is filtered by the vertical direction at the second pass. The first filtering is denoted as:

$$\bar{\boldsymbol{I}}^H_{\boldsymbol{p}} = \frac{1}{K^H_{\boldsymbol{p}}} \sum_{\boldsymbol{q}\in\Omega_H} w^H_{\boldsymbol{p},\boldsymbol{q}} \boldsymbol{I}_{\boldsymbol{q}}, \tag{5}$$

where Ω_H is a subset along horizontal direction of a full kernel. $w^H_{\boldsymbol{p},\boldsymbol{q}}$ and $\bar{\boldsymbol{I}}^H_{\boldsymbol{p}}$ are a weight and an output of the horizontal filtering, respectively, and $K^H_{\boldsymbol{p}}$ is the normalization factor. The second filtering is represented as:

$$\bar{\boldsymbol{I}}^{sp}_{\boldsymbol{p}} = \frac{1}{K^V_{\boldsymbol{p}}} \sum_{\boldsymbol{q}\in\Omega_V} w^V_{\boldsymbol{p},\boldsymbol{q}} \bar{\boldsymbol{I}}^H_{\boldsymbol{q}}. \tag{6}$$

Here, Ω_V is a subset along vertical direction of a full kernel, $w^V_{\boldsymbol{p},\boldsymbol{q}}$ is a weight of the vertical filtering, and $K^V_{\boldsymbol{p}}$ is the normalization factor. $\bar{\boldsymbol{I}}^{sp}_{\boldsymbol{p}}$ is an output of the conventional separable filtering.

The weight of the separable filtering has the relation to the pixel position, the input image, and the parameters; thus, the weights of each direction become the following weighting functions:

$$w^H_{\boldsymbol{p},\boldsymbol{q}} = \mathrm{weight}(\boldsymbol{p}, \boldsymbol{q}, \boldsymbol{I}, \boldsymbol{\beta}, \boldsymbol{\gamma}), \tag{7}$$

Table 1: Difference of the second-pass filtering between the conventional implementation and the proposed one.

	Conventional	Proposed
Guidance	Filtered image	Input image
Filtering type	Direct filtering	Joint filtering
Sigma of range	No change	Compressed

$$w^V_{\boldsymbol{p},\boldsymbol{q}} = \mathrm{weight}(\boldsymbol{p}, \boldsymbol{q}, \bar{\boldsymbol{I}}^H, \boldsymbol{\beta}, \boldsymbol{\gamma}), \tag{8}$$

where $\boldsymbol{\beta}$ is parameter(s) for a domain kernel (e.g., σ_s for the bilateral filter), and $\boldsymbol{\gamma}$ is parameter(s) for a range kernel (e.g., σ_r for the bilateral filter). Note that the first-pass weight of w^H is calculated by using the input image $\boldsymbol{I}$ and the second-pass weight of w^V is calculated by using the *horizontally filtered image* $\bar{\boldsymbol{I}}^H$. The second-pass weight is computed from the filtering image itself; thus, we can use the same function of the first-pass for the second-pass.

3. SWITCHING DUAL KERNELS FOR SEPARABLE FILTER

We propose novel implementation of separable edge-preserving filtering. The conventional implementation [29, 30] utilizes the same kernel function for separable filtering. The drawback of this separable implementation is that vertical filtering (or second pass filtering) results in over-smoothing.

The proposed implementation has an additional kernel for controlling the smoothness at the second-pass filtering and alternates the kernel of the second pass filtering to it. The proposed framework of switching dual kernels based separable filtering (SDK-SF) is depicted in Fig. 1. For the second-pass filtering, we re-define the following equation instead of using Eq. (8):

$$w^V_{\boldsymbol{p},\boldsymbol{q}} = \mathrm{weight}(\boldsymbol{p}, \boldsymbol{q}, \boldsymbol{I}, \boldsymbol{\alpha}\boldsymbol{\beta}, \boldsymbol{\gamma}), \tag{9}$$

where $\boldsymbol{\alpha}, (0 < \alpha_{ii} \leq 1)$ represents a compression ratio of $\boldsymbol{\beta}$, and is a diagonal matrix, which size is $\Gamma \times \Gamma$, where Γ is the dimension of the vector $\boldsymbol{\gamma}$. The main differences between Eq. (8) and (9) are listed in Table 1. We replace the filtered image $\bar{\boldsymbol{I}}$ to the input image $\boldsymbol{I}$ for the range kernel computation. At the conventional second pass, the filtering image is the horizontally filtered image $\bar{\boldsymbol{I}}^H$; thus, we should exploit joint filtering [6]. Moreover, we shorten the width of the range kernel or suppress the weight by using the scaling parameter $\boldsymbol{\alpha}$. As a result, we can protect images from over-smoothing at the second pass filtering stage.

The detailed representation of the bilateral filter is as follows. The conventional vertical weight is:

$$w^{C:V:bi}_{\boldsymbol{p},\boldsymbol{q}} = \exp\left(\frac{-(\|\boldsymbol{p}-\boldsymbol{q}\|_2^2}{2\sigma_s^2}\right)\exp\left(\frac{-\|\bar{\boldsymbol{I}}^H_{\boldsymbol{p}} - \bar{\boldsymbol{I}}^H_{\boldsymbol{q}}\|_2^2}{2\sigma_r^2}\right). \tag{10}$$

Moreover, the proposed implementation is alternated by:

$$w^{P:V:bi}_{\boldsymbol{p},\boldsymbol{q}} = \exp\left(\frac{-(\|\boldsymbol{p}-\boldsymbol{q}\|_2^2}{2\sigma_s^2}\right)\exp\left(\frac{-\|\boldsymbol{I}_{\boldsymbol{p}} - \boldsymbol{I}_{\boldsymbol{q}}\|_2^2}{2\alpha^2\sigma_r^2}\right). \tag{11}$$

Fig. 2: The bilateral filtering results ("flower"): from left to right, input image, naïve, conventional separable implementation, and proposed separable implementation. The filtering parameters are $\sigma_c = 65, \sigma_s = 16.33, r = 49$, and $\alpha = 0.8$ for the proposed. The computational time is 450 ms (naïve), 16 ms (conventional) and 17 ms (proposed).

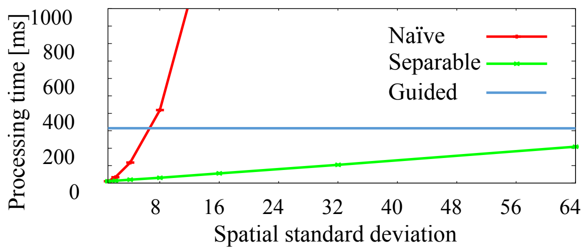

Fig. 3: Computational time versus spatial standard deviation. The kernel radius is set to 3σ. Separable is SDK-SF. Inputs are 1 megapixel (1024×1024) color images.

The representation of the dual bilateral filter and that of the non-local means filter are almost the same and easily derived; thus, we omit these equations from this paper.

4. EXPERIMENTAL RESULTS

In this section, we verify the proposed separable filtering of SDK-SF for the bilateral filter, the non-local means filter, and the dual bilateral filter. We compare the SDK-SF with the conventional separable filters [29, 30]. The results of the bilateral filter are shown in Figs. 2, 3, 4, and these of the non-local means filter are revealed in Figs. 5, 7. The results of the dual bilateral filter is indicated in Fig. 6. Fig. 8 shows computational time for various edge-preserving filters.

Figure 2 shows the bilateral filtering results. The result of the SDK-SF is visually similar to one of the naïve implementation while the conventional implementation has over-smoothing effects along the vertical direction.

Figure 3 shows the computational time of the naïve implementation, the SDK-SF of bilateral filtering and guided filtering [9] that is the representative of constant time filtering. The computational time of the naïve and the SDK-SF is monotonically increasing though the increase of the separable implementation is moderate. The SDK-SF is faster than the guided filter, which is known as the most efficient edge-preserving filter. The cross point between the SDK-SF and the guided filter is at the large kernel width (over 192 pixels with the 1024 image). Note that we omit the conventional implementation because the computational time is almost the same as the proposed one (See Fig. 8).

Figure 4 shows the accuracy of the filtering. We compare the naïve with the SDK-SF of bilateral filtering by using peak

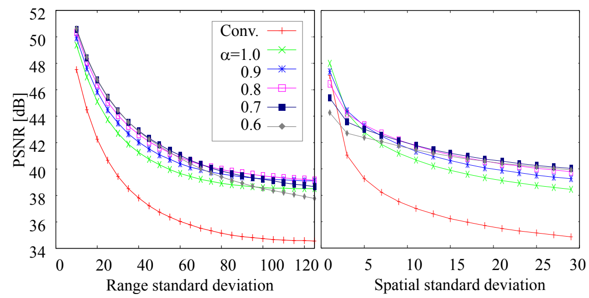

Fig. 4: PSNR versus standard deviation of range kernel (left) and spatial kernel (right). In the left case, σ_s is fixed (16.33), and the kernel radius is 49 (3σ). In the right case, σ_r is fixed (60.0). "Conv." is the conventional implementation, and α is the parameter for the range kernel. The plots are averages of 24 images in Kodak True Color Image Dataset.

signal noise ratio (PSNR) in the Y channel. When we exploit joint filtering for the second pass ($\alpha = 1$), the accuracy is largely improved. Moreover, the accuracy is also improved additionally by adjusting the parameter α (0.8 is the best or the second best for α in the experiments of Fig.4.). Results of the other filters are omitted due to the room of the space. The tendencies are almost the same as the bilateral filter.

Figure 5 shows results of denoising by using the non-local means filter. The SDK-SF implementation has the second highest performance, and the computational cost is almost the same as the conventional separable implementation. The SDK-SF is similar to the naïve implementation, while the convention has vertical streaking noises. Figure 7 shows the denoising results with various noise levels ($\sigma = 10, 20, 30$). The SDK-SF has the second best performance.

Figure 6 shows results of disparity map refining from a coded disparity map. In this experiment, the disparity map and the associated RGB image are coded by JPEG with the quality factor of 50, and then the coded disparity map is filtered by the dual bilateral filter. The range kernel is introduced by the disparity map and the RGB image. The result shows that the SDK-SF is the second best for the disparity map refinement. The filter is useful for real-time depth image based rendering [20] with disparity map coding [18].

The computational time of each edge-preserving filtering is shown in Fig 8. We use C++ and optimize it by using SSE vectorization and multi-core-parallelization with Intel thread-

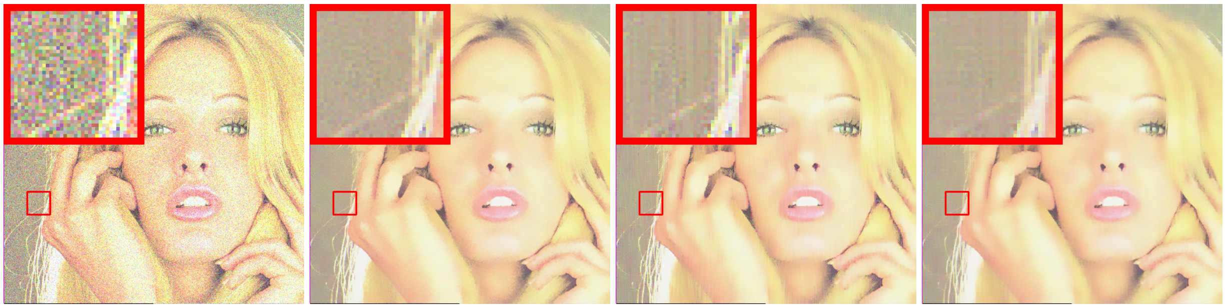

Fig. 5: The non-local means filtering results of the image ("tiffany"): from left to right, noisy image ($\sigma = 30, 512 \times 512$), naïve, conventional separable implementation, and SDK-SF. PSNRs are 19.52, 27.59, 27.82, 28.13 respectively. The computational time of naïve, the conventional and SDK-SF are 33.2 ms, 7ms, 7.2ms, respectively.

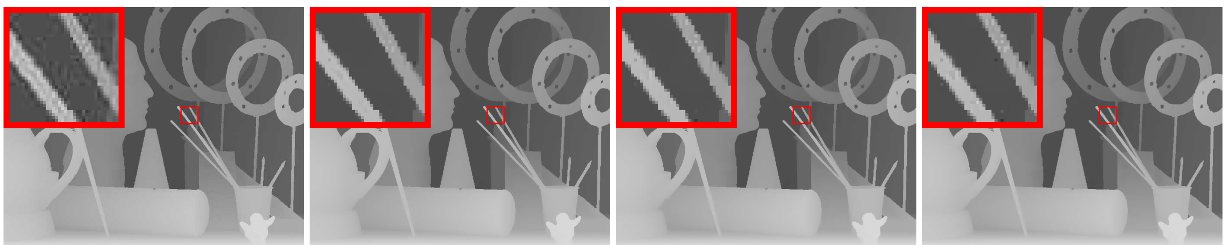

Fig. 6: The dual bilateral filtering results of the disparity map ("art"): from left to right, coded disparity map (JPEG quality factor = 50), naïve, conventional separable implementation, and SDK-SF. The values of ratio of bad pixels [34], are 11.68, 2.25, 2.62, 2.39 respectively. Here, the error threshold set to 1.0.

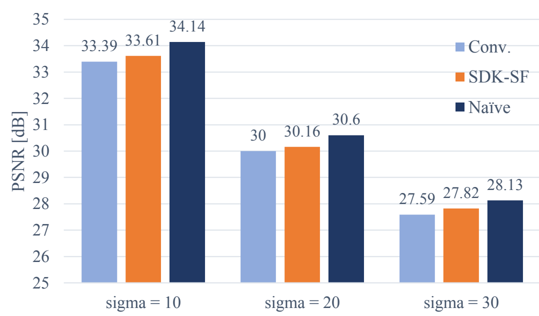

Fig. 7: Denoising results of non-local means filtering with respect to various noise levels ($\sigma = 10, 20, 30$).

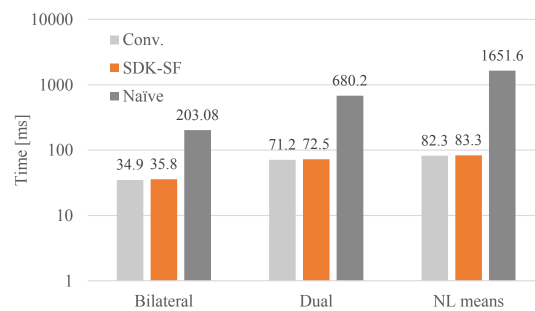

Fig. 8: Computational time of each edge-preserving filter. Image resolution is 1024×1024 color image. The kernel radius is 24×24. The guidance image is color for dual bilateral filtering. Patch size for non-local means filtering is 5×5.

ing building blocks (TBB). CPU is Intel Xeon X5690 3.47 GHz (dual-CPUs), OS is Windows 7 64 bit, and the compiler is Visual Studio 2012.

5. CONCLUSION

In this paper, we proposed new separable implementation for edge-preserving filtering to improve the filtering accuracy. We called the implementation switching dual kernels based separable filtering (SDK-SF). Using joint filtering and adjusting range weights for the second pass filtering, we can suppress over-smoothing effects and streaking noises in the separable edge-preserving filtering. To verify the effectiveness of the SDK-SF, we confirm three types of edge-preserving filtering, such as bilateral filtering, dual bilateral filtering, and non-local means filtering. Experimental results showed that the separable implementation accelerates the edge-preserving filters, and SDK-SF has higher accuracy than the conventional method while the computational time is almost the same as the conventional separable filtering implementation.

Limitation of the separable filtering is that the filter does not suit to complex-textured regions. The filter only travels through horizontal and vertical directions; thus, the smoothing effect cannot be propagated over-striding multiple edges. The limitation is same as the conventional separable implementation.

6. REFERENCES

[1] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *International Conference on Computer Vision (ICCV)*, 1998, pp. 839–846.

[2] A. Buades, B. Coll, and J. M. Morel, "A non-local algorithm for image denoising," in *Computer Vision and Pattern Recognition (CVPR)*, 2005, pp. 60–65.

[3] F. Durand and J. Dorsey, "Fast bilateral filtering for the display of high-dynamic-range images," *ACM Trans. on Graphics*, vol. 21, no. 3, pp. 257–266, 2002.

[4] S. Bae, S. Paris, and F. Durand, "Two-scale tone management for photographic look," *ACM Trans. on Graphics*, vol. 25, no. 3, pp. 637–645, 2006.

[5] R. Fattal, M. Agrawala, and S. Rusinkiewicz, "Multiscale shape and detail enhancement from multi-light image collections," *ACM Trans. on Graphics*, vol. 26, no. 3, 2007.

[6] G. Petschnigg, M. Agrawala, H. Hoppe, R. Szeliski, M. Cohen, and K. Toyama, "Digital photography with flash and no-flash image pairs," *ACM Trans. on Graphics*, vol. 23, no. 3, pp. 664–672, 2004.

[7] E. Eisemann and F. Durand, "Flash photography enhancement via intrinsic relighting," *ACM Trans. on Graphics*, vol. 23, no. 3, pp. 673–678, 2004.

[8] J. Kopf, M. Cohen, D. Lischinski, and M. Uyttendaele, "Joint bilateral upsampling," *ACM Trans. on Graphics*, vol. 26, no. 3, 2007.

[9] K. He, J. Shun, and X. Tang, "Guided image filtering," in *European Conference on Computer Vision (ECCV)*, 2010.

[10] N. Kodera, N. Fukushima, and Y. Ishibashi, "Filter based alpha matting for depth image based rendering," in *Visual Communications and Image Processing (VCIP)*, Nov. 2013.

[11] K. He, J. Sun, and X. Tang, "Single image haze removal using dark channel prior," in *Computer Vision and Pattern Recognition (CVPR)*, 2009, pp. 2341–2353.

[12] K. J. Yoon and I. S. Kweon, "Adaptive support-weight approach for correspondence search," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 28, no. 4, pp. 650–656, 2006.

[13] A. Hosni, C. Rhemann, M. Bleyer, C. Rother, and M. Gelautz, "Fast cost-volume filtering for visual correspondence and beyond," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 35, no. 2, pp. 504–511, 2013.

[14] Q. Yang, R. Yang, J. Davis, and D. Nister, "Spatial-depth super resolution for range images," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007.

[15] S. Fujita, T. Matsuo, N. Fukushima, and Y. Ishibashi, "Cost volume refinement filter for post filtering of visual corresponding," in *Proc. SPIE*, 2015.

[16] T. Matsuo, N. Fukushima, and Y. Ishibashi, "Weighted joint bilateral filter with slope depth compensation filter for depth map refinement," in *International Conference on Computer Vision Theory and Applications (VISAPP)*, Feb. 2013, pp. 300–309.

[17] T. Matsuo, S. Fujita, N. Fukushima, and Y. Ishibashi, "Efficient edge-awareness propagation via single-map filtering for edge-preserving stereo matching," in *Proc. SPIE*, 2015.

[18] S. Liu, P. Lai, D. Tian, and C. W. Chen, "New depth coding techniques with utilization of corresponding video," *IEEE Trans. on Broadcasting*, vol. 57, no. 2, pp. 551–561, June 2011.

[19] N. Fukushima, T. Inoue, and Y. Ishibashi, "Removing depth map coding distortion by using post filter set," in *International Conference on Multimedia and Expo (ICME)*, July 2013.

[20] Y. Mori, N. Fukushima, T. Yendo, T. Fujii, and M. Tanimoto, "View generation with 3d warping using depth information for ftv," *Signal Processing: Image Communication*, vol. 24, no. 1-2, pp. 65 – 72, 2009.

[21] F. Porikli, "Constant time o(1) bilateral filtering," in *Computer Vision and Pattern Recognition (CVPR)*, 2008.

[22] Q. Yang, K. H. Tan, and N. Ahuja, "Real-time o(1) bilateral filtering," in *Computer Vision and Pattern Recognition (CVPR)*, 2009, pp. 557–564.

[23] S. Paris and F. Durand, "A fast approximation of the bilateral filter using a signal processing approach," *International Journal of Computer Vision*, vol. 81, no. 1, pp. 24–52, 2009.

[24] A. Adams, N. Gelfand, J. Dolson, and M. Levoy, "Gaussian kd-trees for fast high-dimensional filtering," *ACM Trans. on Graphics*, vol. 28, no. 3, 2009.

[25] A. Adams, J. Baek, and M. A. Davis, "Fast high-dimensional filtering using the permutohedral lattice," *Computer Graphics Forum*, vol. 29, no. 2, pp. 753–762, 2010.

[26] E. S. L. Gastal and M. M. Oliveira, "Adaptive manifolds for real-time high-dimensional filtering," *ACM Trans. on Graphics*, vol. 31, no. 4, 2012.

[27] E. S. L. Gastal and M. M. Oliveira, "Domain transform for edge-aware image and video processing," *ACM Trans. on Graphics*, vol. 30, no. 4, 2011.

[28] Q. Yang, "Recursive bilateral filtering," in *European Conference on Computer Vision (ECCV)*, 2012, pp. 399–413.

[29] T. Q. Pham and L. J. V. Vliet, "Separable bilateral filtering for fast video preprocessing," in *International Conference on Multimedia and Expo (ICME)*, 2005.

[30] Y. S. Kim, H. Lim, O. Choi, K. Lee, J.D.K. Kim, and J.D. Kim, "Separable bilateral nonlocal means," in *International Conference on Image Processing (ICIP)*, Sep. 2011, pp. 1513–1516.

[31] P. Kornprobst and J. Tumblin, *Bilateral filtering: Theory and applications*, Now Publishers Inc., 2009.

[32] I. T. Butt and N. M. Rajpoot, "Multilateral filtering: A novel framework for generic similarity-based image denoising," in *International Conference on Image Processing (ICIP)*, 2009, pp. 2981–2984.

[33] S. Treitel and J.L. Shanks, "The design of multistage separable planar filters," *IEEE Trans. on Geoscience Electronics*, vol. 9, no. 1, pp. 10–27, Jan 1971.

[34] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International Journal of Computer Vision*, vol. 47, no. 1-3, pp. 7–42, Apr. 2002.