Cost Volume Refinement Filter for Post Filtering of Visual Corresponding

Shu Fujita, Takuya Matsuo, Norishige Fukushima, Yutaka Ishibashi

Nagoya Institute of Technology

ABSTRACT

In this paper, we propose a generalized framework of cost volume refinement filtering for visual corresponding problems. When we estimate a visual correspondence map, e.g., depth map, optical flow, segmentation and so on, the estimated map often contains a number of noises and blurs. One of the solutions for this problem is post filtering. Edge-preserving filtering, such as joint bilateral filtering, can remove the noises, but it causes blurs on object boundaries at the same time. As an approach to remove noises without blurring, there is cost volume refinement filtering (CVRF) that is an effective solution for the refinement of such labeling of correspondence problems. There are some papers that propose several methods categorized into CVRF for various applications. These methods use various reconstructing metrics functions, which are L1 norm, L2 norm or exponential function, and various edge-preserving filters, which are joint bilateral filtering, guided image filtering and so on. In this paper, we generalize these factors and add range-spacial domain resizing factor for CVRF. Experimental results show that our generalized formulation outperform the conventional approaches, and also show what the format of CVRF is appropriate for various applications of stereo matching and optical flow estimation.

Keywords: cost volume, refinement filter, visual corresponding, depth map, optical flow

1. INTRODUCTION

Recently, basic image processing with depth maps, optical flows or segment images, e.g., pose estimation, object detection, object tracking and free viewpoint video rendering, attracts attentions. If we want high-quality performance in such applications, the accuracy of the maps is required.

To obtain the accurate corresponding maps, there are two main methods: optimization and filtering refinement. The optimization methods can accurately compute the corresponding maps, but its computational cost is usually high. In the case of the refinement methods, while the accuracy is a little lower than the optimization methods, its computational cost is less than the optimization methods. The refinement methods are still not investigated adequately; hence, we can still hope the improvement of the performance. Moreover, the refinement methods can also improve the results of the optimization methods.

Estimation of depth maps, optical flows or segment images can be defined by a discrete label-based problem. A cost volume refinement filter is one of the effective approaches to solve this problem by refining the labels. Various methods using the cost volume refinement filter are actively proposed in.[1–5] However, the cost volume refinement filter is individually formulated using different methods by the previous works (e.g., metrics for building a cost volume and filters for refining the cost volume). Furthermore, there is no comparison among these methods used for the processing of the cost volume filter in the previous works.

Therefore, we generalize the cost volume refinement filter and compare the performance of each format. Experimental results show that we show the best combination of the cost volume refinement filter for various applications, e.g., depth map, optical flow estimation and its dynamic range or resolution up-sampling.

The rest of this paper is organized as follows. Section 2 describes related works. A generalized cost refinement volume filter is defined in Sec. 3. In Sec. 4, experimental results are shown, and appropriate methods for the cost volume filter are discussed. Section 5 concludes this paper.

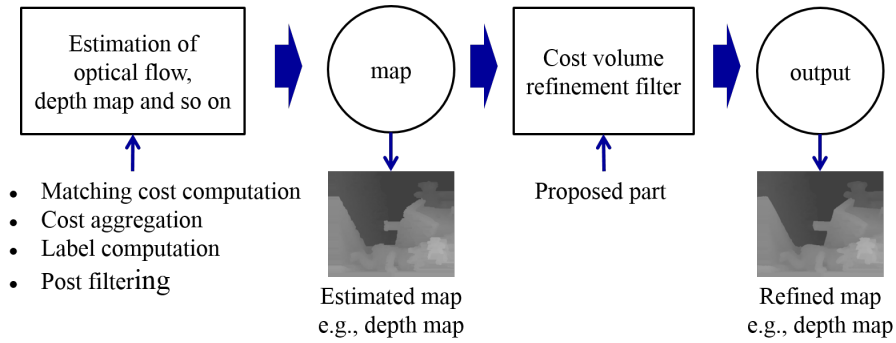


Figure 1: Procedure to obtain high-quality outputs by cost volume refinement filtering.

2. RELATED WORKS

The processing part of the cost volume refinement filter (CVRF) is shown in Fig. 1. CVRF is located in a post processing part of visual corresponding problems. CVRF requires a corresponding map and performs cost computation, cost aggregation and label computation.[5] In this section, we describe what method for the processing in CVRF is used in previous works. Especially, we introduce three typical methods.

A. Hosni *et al.*[1] showed a method that is applied the cost volume refinement filter to estimation methods for discrete labeling problems. They constructed a cost volume at the stage of the matching cost computation. In this time, the cost is computed by L1 norm. The cost volume is refined by using the guided filter.[6]

Q. Yang *et al.*[2] applied the cost volume refinement filter to the enhancement of resolution of a range image. This method iteratively refines a cost volume of the range image which is up-sampled from the low-resolution range image. They compute the cost by L2 norm when they build the cost volume. In filtering the cost volume, the bilateral filter[7] is used.

D. Min *et al.*[4] proposed the weighted mode filtering using joint histograms. The joint histogram is related to the cost volume filter as reported by them.[4] Thus, we regard the weighted mode filtering as one of the cost volume filter. The methods of constructing and refining the cost volume are the exponential function and the joint/cross bilateral filter,[8,9] respectively.

3. GENERALIZED COST VOLUME REFINEMENT FILTER

As mentioned above, a number of methods using CVRF, and they are effective for wide applications as shown in the previous works. However, the appropriate format differs according to applied applications. In addition, each filter of CVRF used in the previous works has not a general format because they are formulated individually. Therefore, we firstly generalize CVRF in order to apply various formats.

3.1 Cost Volume Refinement Filter

There are three main steps in CVRF. They are building a cost volume, refining cost slices, merging the cost volume. The overview of CVRF is shown in Fig. 2. In this section, we describe and generalize the each process.

3.1.1 Building cost volume

First, we discuss the process to build a cost volume. The cost volume V consists of N slices of a cost slice V_n ($n \in \{0, \dots, N-1\}$) that is made in each label. We can apply various metrics to this process; hence, we define $V_n(\boldsymbol{p})$ that is a pixel value on the pixel $\boldsymbol{p}$ in the cost slice V_n as follows:

$$V_n(\boldsymbol{p}) = L(n, I(\boldsymbol{p}), \tau) \quad 0 \leq n \leq N-1 \tag{1}$$

where L is a cost function for building the cost slice, n is a label value that each cost slice has, I is an estimated image, τ is a truncation value. We should select a monotonically increasing function as the cost function L,

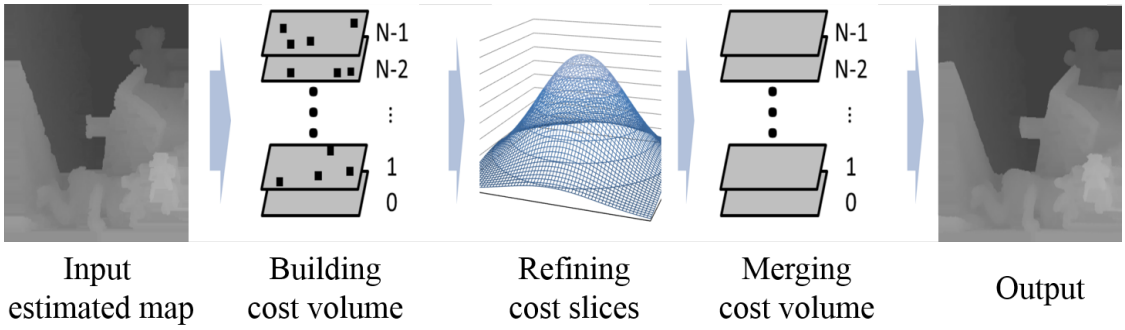


Figure 2: Overview of cost volume refinement filtering.

although the function has various types as shown in Tab. 1. In this paper, we compare L1 norm, L2 norm and the exponential function as the representatives. These functions compute costs as follows:

$$L_{L1}(n, I(\boldsymbol{p}), \tau) = \frac{1}{\tau} \min(||n - I(\boldsymbol{p})||_1, \tau) \tag{2}$$

$$L_{L2}(n, I(\boldsymbol{p}), \tau) = \frac{1}{\tau^2} \min(||n - I(\boldsymbol{p})||_2, \tau^2) \tag{3}$$

$$L_{exp}(n, I(\boldsymbol{p}), \tau) = 1 - \exp\left(-\frac{\|n - I(\boldsymbol{p})\|^2}{2\tau^2}\right), \tag{4}$$

where L_{L1}, L_{L2} and L_{exp} are L1 norm, L2 norm and exponential function, respectively. Also, $||\cdot||_1$ and $||\cdot||_2$ denote L1 norm and L2 norm, respectively.

3.1.2 Refining cost slices

Next, we explain the process for refining the cost slices. Since the cost slices after building the cost volume usually contain noises, we refine them by filtering:

$$V'_n(\boldsymbol{p}) = \sum_{\boldsymbol{s} \in S(\boldsymbol{p})} f(\boldsymbol{p}, \boldsymbol{s}) V_n(\boldsymbol{s}), \tag{5}$$

where V' is a refined cost volume, $S(\boldsymbol{p})$ is a set of support pixel $\boldsymbol{s}$ around $\boldsymbol{p}$ and f is a filtering weight function. We can refine them by using any filters if the filters have the effect of noise reduction, but a recommended type of the filter is edge-preserving filters.[6,8–15] The reason is that we can suppress mixture of costs around regions of object boundaries. We show the examples of the filtering method in Tab. 1.

In addition, the performance of refining the cost slices becomes high by using a weight map. The authors find that it is effective to use a weight map as representing pixel reliabilities for the depth map refinement reported in,[16] and this is an extension version. Instead of filtering corresponding maps directly, we filter cost slices. This refinement process as follows:

$$V'_n(\boldsymbol{p}) = \sum_{\boldsymbol{s} \in S(\boldsymbol{p})} f(\boldsymbol{p}, \boldsymbol{s}) M(\boldsymbol{s}) V_n(\boldsymbol{s}). \tag{6}$$

There are also various types for the weight map M as shown in Tab. 1. Especially, the trilateral weight map has good performance.[16] Due to this, we use the trilateral weight map in this paper. The trilateral weight map is computed as follows:

$$M(\boldsymbol{p}) = \sum_{s \in S(\boldsymbol{p})} w(\boldsymbol{p}, \boldsymbol{s}) c(I(\boldsymbol{p}), I(\boldsymbol{s})) d(R(\boldsymbol{p}), R(\boldsymbol{s})). \tag{7}$$

Here, R is a guidance image, w, c and d are exponential functions: $\exp(-\frac{\|\boldsymbol{x}-\boldsymbol{y}\|_2}{2\sigma^2_{s/c/d}})$, where $\sigma_s, \sigma_c, \sigma_d$ are spatial, guide color and self value standard deviation, respectively.

Table 1: Examples of method for building cost volume and refining cost slices.

Building cost volume	Refining cost slices	
Cost function	Filter	Weight map
• L1norm function	• Gaussian filter	• Trilateral weight map[16]
• L2norm function	• Joint bilateral filter[8,9]	• LR consistency map[5]
• Exponential function	• Guided filter[6]	• Speckle mask
• Sigmoid function	• Trilateral filter[10]	• Uniqueness map
• Cubed differences function	• Domain transform filter[11]	• Up-sampling mask

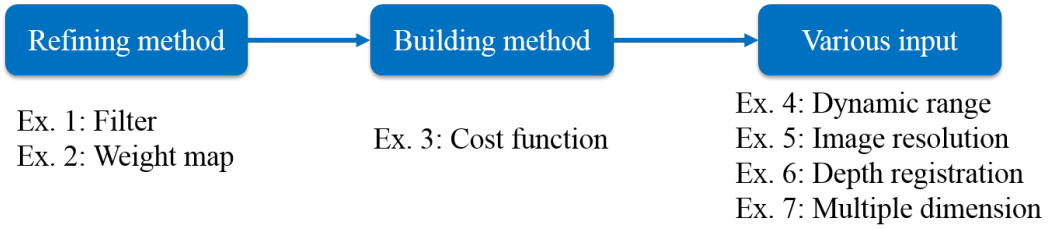


Figure 3: Overall of our experiments.

3.1.3 Merging cost volume

Finally, we talk about merging the cost volume. After the cost volume is refined, we choose the minimum cost label at a pixel $\boldsymbol{p}$:

$$O(\boldsymbol{p}) = \arg\min_{n} V'_n(\boldsymbol{p}), \tag{8}$$

where O is the output image. After that, we can additionally conduct sub-pixel interpolation for increasing the sub-pixel accuracy. In this paper, we apply the quadratic estimator[17] as the sub-pixel interpolation method.

In this way, we can generalize each process of CVRF. We can obtain a refined map when we have performed these processes in all pixels.

3.2 Application

CVRF can be applied to refining various maps, for example, segment image, depth map, optical flow, alpha map[18] and transmission map.[19] The difference of the processing for single channel maps such as segment images and depth maps is the number of the labels. Consequently, the process for the maps are almost the same. On the other hand, various methods for refining multi-channel maps such as optical flow are possible, although we filter each channel in this paper.

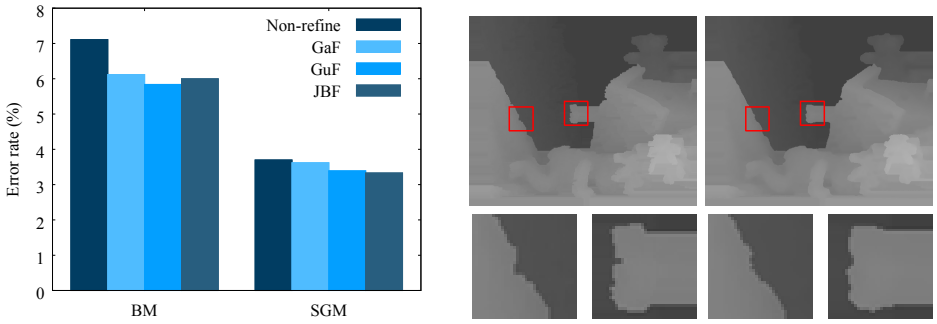
4. EXPERIMENTAL RESULTS

4.1 Experimental Environment

In this experiment, we refine depth maps as a representative of single channel corresponding maps and optical flows as a representative of multi-channel corresponding maps. Note that we estimate depth maps by block matching (BM)[20] and semi-global matching (SGM).[21] The methods for the estimating optical flows are F-TV-L1[22] and Farneback algorithm.[23] The objective evaluation method is error rate.[20,24] It is calculated by the percentage of error pixels of an estimated image. We only evaluate non-occluded regions in this paper.

The definition of the error pixel differs between depth maps and optical flows. In the case of depth maps, difference of pixel values between the input and ground truth depth map has over a threshold value γ ($\gamma = 1$ in our experiments). In the case of the optical flows, difference of vector angles and vector lengths between the input and the ground truth has over threshold values (set to 5.0 and 1.0 in our experiments, respectively). These errors are called angular error (AE) and endpoint error (EE).[24]

We use the Middlebury's data sets[20,24] as input images. For depth maps, "Tsukuba", "Venus", "Teddy" and "Cones" are used. For optical flows, "RubberWhale" and "Grove2" are used. Since we utilize multiple images, we denote their average error rates as our results in each application.



(a) Error rate (%) (b) BM (c) CVRF with JBF

Figure 4: Difference of filter. (a) Result of averaged error rate among 4 datasets. The method used for building a cost volume is L2 norm. The number of the cost slices is 256. (b)-(d) are picked up results of Teddy. (b) Estimated image by BM. (c) Refined image of (b) by CVRF with JBF.

We evaluate the refinement performance of CVRF by conducting six experiments. The overall of our experiments is shown in Fig. 3. Note that we use depth map as the input map in the 1st-5th experiment. Additionally, we add Gaussian noises depending on the experiments in order to assume that depth maps obtained by a depth sensor usually include noises.

In the 1st experiments, we investigate the impact of the refinement performance by using various filters. The 2nd experiments show the effectiveness of presence or absence of a weight map. For refining the cost slices, we use the Gaussian filter (GaF), the guided filter (GuF),[6] the joint bilateral filter (JBF)[8] in this paper. The 3rd experiment is about a cost function for building a cost volume. We use L1 norm, L2 norm and the exponential function as mentioned Sec. 3.1.1. The rest of our experiments focus on the difference of condition of the input map. Especially, we discuss dynamic range and resolution of depth maps and multi-dimensional corresponding map of optical flows. In the experiment of dynamic range, the range is down-sampled and up-sampled to half and double, respectively. In the case of the resolution up-sampling, we up-sample the depth map to original size using the color based depth up-sampling[25] after down-sampling the input depth map to half size using nearest neighbor algorithm.

4.2 Results and Discussions

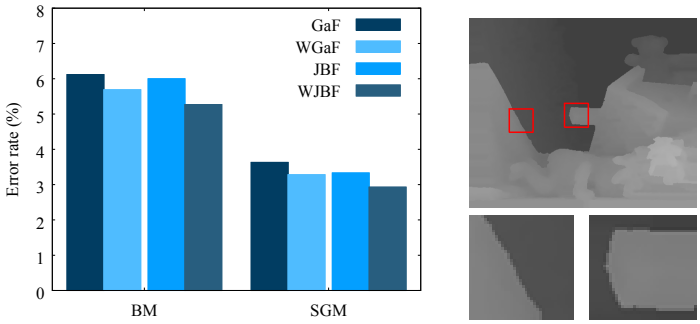
4.2.1 Difference of filter

We first show the experimental results of the difference between filters in Fig. 4. In this regard, the parameters of the filters are experimentally determined in all cases to have the best performance. The amount of improvement is large when we use the edge-preserving filters from Fig. 4 (a). Also, we can confirm that the edges are corrected from Figs. 4 (b), (c). Here, the performances of JBF and GuF are almost the same; hence, we use JBF as the edge-preserving filter following experiments.

4.2.2 Presence or absence of weight map

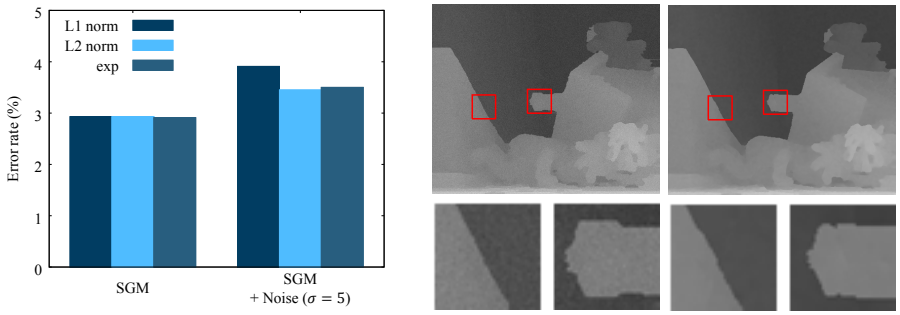
Figure 5 shows the result about the effect of the trilateral weight map. There are four parameters at the trilateral weight map generation. The parameters are $(\sigma_s, \sigma_c, \sigma_d, r) = (50, 4, 6, 50)$.

As compared to presence or absence of the weight map, the difference of the refinement performance is large from Fig. 5 (a). Here, we call JBF and GaF with the trilateral weight map as WJBF and WGaF, respectively. Especially, the amount of improvement is bigger than the edge-preserving filter without the trilateral weight map even if we use WGaF that does not have the effect of edge-preserving. In the refined image, the performance of edge correction becomes high when we compare Fig. 5 (c) with Fig. 5 (b). Thus, we use the trilateral weight map following experiments.



(a) Error rate (%) (b) CVRF with WJBF

Figure 5: Presence or absence of weight map. (a) Result of error rate. The method used for building a cost volume is L2 norm. The number of the cost slices is 256. (b) Refined image of Fig. 4b by CVRF with WJBF.



(a) Error rate (%) (b) SGM & noise (c) CVRF with WJBF

Figure 6: Difference of method for building cost volume. (a) Result of error rate. The filter used for refining the cost slices is WJBF. The number of the cost slices is 256. (b) Noise added image. The error rate is 42.38 %. (c) Refined image of (b) by CVRF using L2 norm with WJBF.

4.2.3 Difference of method for building cost volume

We show the result of the difference of the methods for building the cost volume in Fig. 6. We can confirm that the difference of the amount of improvement is little from Fig. 6 (a) when the input depth map is high quality. However, there is a clear difference when the input includes noises. In using L1 norm for building the cost volume, the amount of improvement becomes low relative to the other methods*. From this result, L1 norm is not appropriate for this process.

4.2.4 Dynamic range up or down-sampling

As shown in Fig. 7, there is little difference when $\gamma = 1$. On the other hand, if we measure the amount of improvement in the sub-pixel level, it becomes low when we down-sample the dynamic range. Consequently, we should not down-sample the dynamic range if we want the accurate result. Moreover, the effect is almost nothing in the case of the up-sampling the dynamic range, so we had better not up-sampling it.

4.2.5 Resolution up-sampling and noisy up-sampling

The result of refinement for an up-sampled depth map shows in Fig. 8. In the case of up-sampling not including noises, WJBF is better. However, the amount of improvement of WGaF is lower than WJBF when we up-sample the noisy depth map. We consider the reason that noises have been extended by up-sampling, and hence WJBF can not reduce the noises relative to WGaF because of edge-preserving effect. Actually, we can confirm that the performance of edge correction of WJBF is better than WGaF from Figs. 8 (b), (c), (d). Therefore, we should reduce the noises before up-sampling when we up-sample a depth map including noises.

*This tendency is almost the same as the estimated depth map by BM.

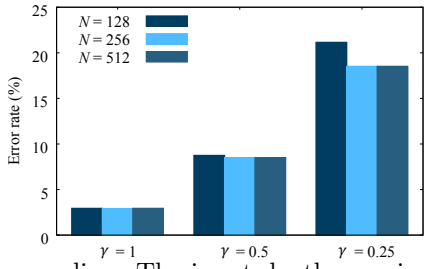
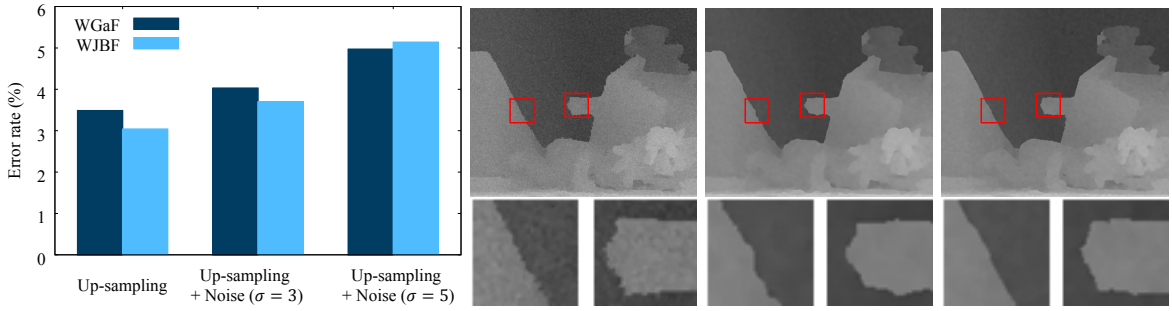


Figure 7: Dynamic range up or down-sampling. The input depth map is estimated by SGM. The method used for building a cost volume is L2 norm. The filter used for refining the cost volume is WJBF.



(a) Error rate (%) (b) Noise & up-sample (c) CVRF with WGaF (d) CVRF with WJBF

Figure 8: Resolution up-sampling and noisy up-sampling. The input depth map is estimated by SGM. The noise is added before up-sampling. (a) Result of error rate. The method used for building a cost volume is L2 norm. The number of the cost slices is 256. (b) Noise + Up-sampled image. The error rate is 41.98 %. (c) - (d) Refined images of (b) by CVRF with WGaF and WJBF, respectively.

4.2.6 Depth map registration for unstructured pixels from depth sensor

In this section, we demonstrate the effect of the weight map for unstructured depth map registration. When we capture a depth map and its associated RGB image captured with a depth sensor, positions and resolution is different between RGB and depth camera. Registering the depth map to the RGB image coordinates, we project pixels in the depth map to the RGB image. After that, the depth map is sparsely mapped to the RGB image coordinates. Three factors make the registered-depth-map sparse; difference of camera position, difference of image resolution and lens distortion, unreliable pixels. The position of the missing pixels is unstructured manner; thus a suitable interpolation is required for making dense depth map.

CVRF whose weight map indicates missing pixels can interpolate depth map well. Figure 9 shows an example of the depth map registration by using a depth sensor of Kinect V2. The resolution of image is 1920×1080 and of depth is 512×424. The result shows that the registered depth map is well construed without the outside of the map. The missing pixels in outside are caused by the difference between the field of view of the cameras and the distortion of the cameras.

4.2.7 Optical flow refinement

Here, we discuss the effect of the optical flow refinement. Figure 10 shows the results of optical flow refinements. As a whole, we can confirm that the optical flow refinement by CVRF is effective. The notable point is that blurs on the object boundaries are removed as shown in Figs. 10 (c), (d), (e). However, in common with the noisy up-sampling part, the performance of WGaF becomes higher than WJBF when the quality of the input flow is low. The reason is almost same as the noisy up-sampling part.

5. CONCLUSION

In this paper, we generalized a cost volume refinement filter (CVRF) and evaluated the performance of CVRF applied by various methods. Experimental results showed that CVRF is effective for various applications. Moreover, we demonstrated the appropriate methods for the cost volume filter by comparing each method.

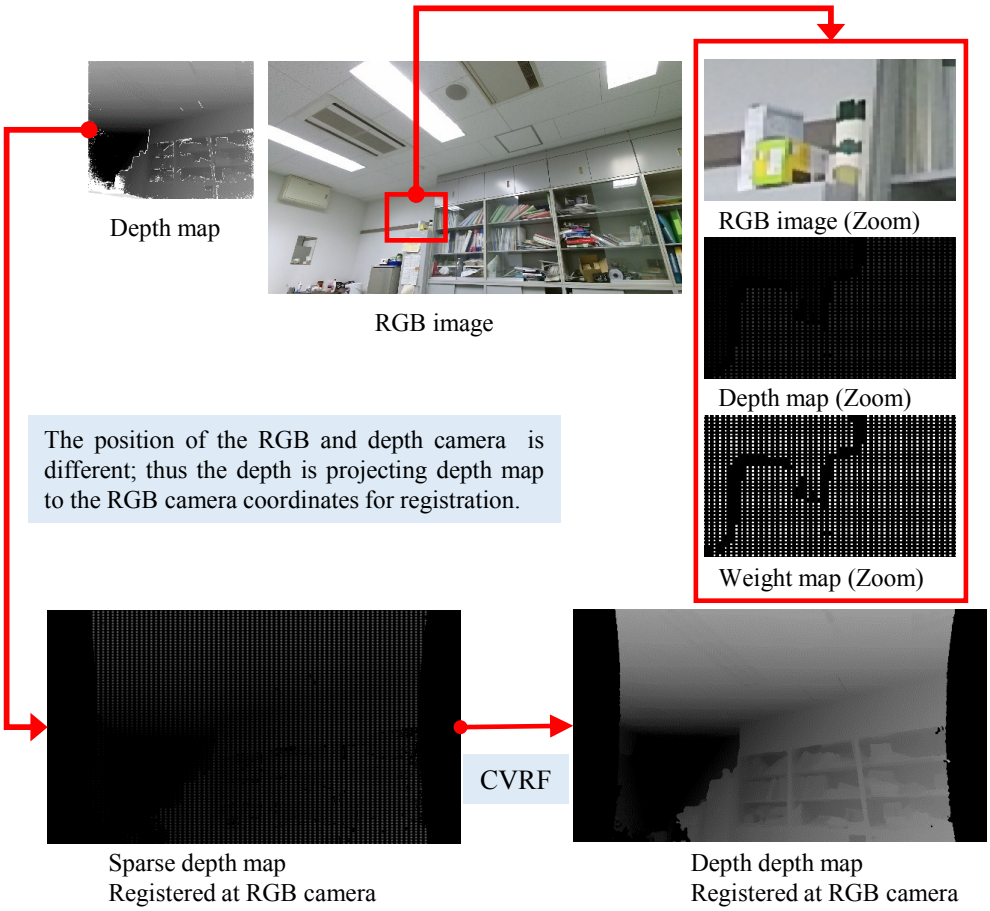


Figure 9: Unstructured depth map up-sampling.

Although we only used the trilateral weight map as a weight map in this paper, there are the other weight maps. Therefore, we consider that our future work investigates the difference of performance between weight maps.

REFERENCES

[1] Hosni, A., Rhemann, C., Bleyer, M., Rother, C., and Gelautz, M., "Fast cost-volume filtering for visual correspondence and beyond," *IEEE Trans. on Pattern Analysis and Machine Intelligence* **35**(2), 504–511 (2013).

[2] Yang, Q., Yang, R., Davis, J., and Nistér, D., "Spatial-depth super resolution for range images," in *Proc. CVPR*, 1–8 (2007).

[3] Yang, Q., Ahuja, N., Yang, R., Tan, K., Davis, J., Culbertson, B., Apostolopoulos, J., and Wang, G., "Fusion of median and bilateral filtering for range image upsampling," *IEEE Trans. on Image Processing* **22**(12), 4841–4852 (2013).

[4] Min, D., Lu, J., and Do, M. N., "Depth video enhancement based on weighted mode filtering," *IEEE Trans. on Image Processing* **21**(3), 1176–1190 (2012).

[5] Yang, Q., "A non-local cost aggregation method for stereo matching," in *Proc. CVPR*, 1402–1409 (2012).

[6] He, K., Shun, J., and Tang, X., "Guided image filtering," in *Proc. ECCV*, 1–14 (2010).

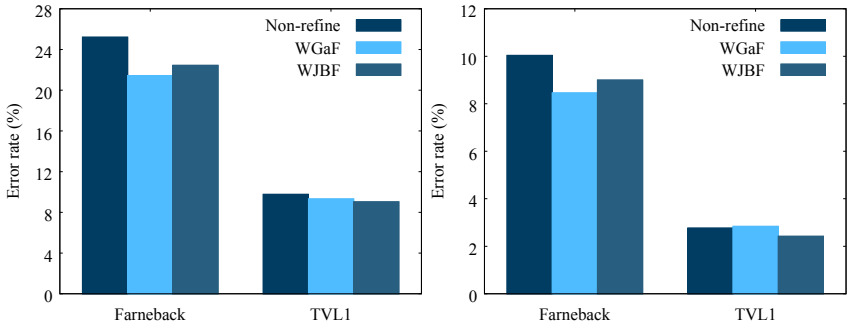
[7] Tomasi, C. and Manduchi, R., "Bilateral filtering for gray and color images," in *Proc. ICCV*, 839–846 (1998).

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

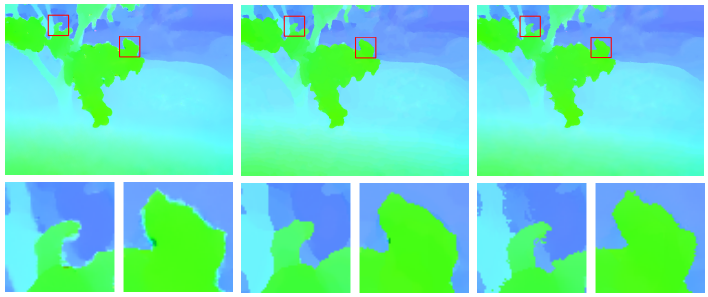
[9] Eisemann, E. and Durand, F., "Flash photography enhancement via intrinsic relighting," *ACM Trans. on Graphics* **23**(3), 673–678 (2004).

[10] Mueller, M., Zilly, F., and Kauff, P., "Adaptive cross-trilateral depth map filtering," in *Proc. 3DTV-Con*, 1–4 (2010).

[11] Gastal, E. S. L. and Oliveira, M. M., "Domain transform for edge-aware image and video processing," *ACM Trans. on Graphics* **30**(4) (2011).



(a) AE (b) EE



(c) TVL1 (d) CVRF with WGaF (e) CVRF with WJBF

Figure 10: Optical flow refinement. (a) - (b) Results of averaged error rate about AE and EE among 2 datasets. The method used for building a cost volume is L2 norm. The number of the cost slices is 128. (c)-(e) are picked up results of Grove2(c). (c) Estimated image by TVL1. (d) - (e) Refined images of (c) by CVRF with WGaF and WJBF, respectively.

[12] Fattal, R., "Edge-avoiding wavelets and their applications," *ACM Trans. on Graphics* **28**(3) (2009).

[13] Paris, S., Hasinoff, W., and Kautz, J., "Local laplacian filters: Edge-aware image processing with a laplacian pyramid," *ACM Trans. on Graphics* **30**(4) (2011).

[14] Pham, T. Q. and Vliet, L. J. V., "Separable bilateral filtering for fast video preprocessing," in *Proc. ICME*, (2005).

[15] Fukushima, N., Fujita, S., and Ishibashi, Y., "Switching dual kernels for separable edge-preserving filtering," in *Proc. ICASSP*, (2015).

[16] Matsuo, T., Fukushima, N., and Ishibashi, Y., "Weighted joint bilateral filter with slope depth compensation filter for depth map refinement," in *Proc. VISAPP*, 300–309 (2013).

[17] Dvornychenko, V., "Bounds on (deterministic) correlation functions with application to registration," *IEEE Trans. on Pattern Analysis and Machine Intelligence* **5**(2), 206–213 (1983).

[18] Rhemann, C., Rother, C., Wang, J., Gelautz, M., Kohli, P., and Rott, P., "A perceptually motivated online benchmark for image matting," in *Proc. CVPR*, 1826–1833 (2009).

[19] He, K., Sun, J., and Tang, X., "Single image haze removal using dark channel prior," in *Proc. CVPR*, 2341–2353 (2009).

[20] Scharstein, D. and Szeliski, R., "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International Journal of Computer Vision* **47**(1), 7–42 (2002).

[21] Hirchmüller, H., "Stereo processing by semi-global matching and mutual information," *IEEE Trans. on Pattern Analysis and Machine Intelligence* **30**(2), 328–341 (2008).

[22] Wedel, A., Pock, T., Braun, J., Franke, U., and D.Cremers, "Duality tv-l1 flow with fundamental matrix prior," in *Proc. International Conference on Image and Vision Computing New Zealand*, 1–6 (2008).

[23] Farneback, G., "Two-frame motion estimation based on polynomial expansion," in *Proc. Scandinavian Conference on Image Analysis*, 363–370 (2003).

[24] Baker, S., Scharstein, D., Lewis, J. P., Roth, S., Black, M. J., and Szeliski, R., "A database and evaluation methodology for optical flow," *International Journal of Computer Vision* **92**(1), 1–31 (2011).

[25] Wildeboer, M., Yendo, T., Tehrani, M., Fujii, T., and Tanimoto, M., "Color based depth up-sampling for depth compression," in *Proc. PCS*, 170–173 (2010).