Matting-Based Stereo Refinement for Computational Photography

Dongwei Liu and Reinhard Klette
The *.enpeda..* Project, Department of Computer Science
The University of Auckland, New Zealand
dliu697@aucklanduni.ac.nz

ABSTRACT

We present a method for refining disparity maps generated by a stereo matcher for the purpose of computational photography. We automatically fill holes in disparity maps, remove noisy artefacts, and enhance visible object geometries based on available disparity and image data, for the purpose of generating visually appealing depth representations. The key idea is that we use image features (e.g. edges) of the base image (say, the left image of the stereo pair) for enhancing the corresponding depth map. To achieve this, we analyse the base image by spectral matting, and then revise disparity values by a weighted median filter. Experiments show that our method is able to fill holes (i.e. pixels where depth information is unavailable), to revise inaccurate object edges, and to remove speckle noise and invalid step-edges from the given depth information. Besides photo editing, results provided by our method can also be used for image segmentation or object detection.

Categories and Subject Descriptors

I.4.8 [**Image Processing and Computer Vision**]: Scene Analysis

Keywords

Stereo vision, Stereo refinement, Matting, Computational photography

1. INTRODUCTION

Depth information encoded in images is widely used in computer vision or graphics for image segmentation, object detection, 3D reconstruction, 3D visualisation, and other tasks. Various sensors are developed for estimating depth in the real world, such as *binocular stereo* [2, 3], structured lighting as used in the *Kinect* [8], *depth from defocus* (DFD) [14, 15], a *light field camera* [13], 3D laser imaging systems [1], or just GPS and existing georeferenced models [9]. Among these sensors, binocular stereo is the dominant method used for outdoor scenes, and the underlying methodology is close to human visual cognition.

We intend to use generated depth maps for computational photography; for example, changing illumination, simulating fog effect or out-of-focus blur. Stereo vision provides imperfect results for various reasons, even when using a stereo matcher such as iSGM [4].[1] We state here just three of such reasons (see Figure 1):

First, depth values for some pixels in the base image are unavailable, represented by *holes* in the depth map. Those holes can be due to *occlusion*, which is inherent for stereo vision, or to low confidence in stereo matching results at those pixels (see stereo confidence measures in [6]). Such holes can be tolerated to some extent for object detection or distance measurement tasks, but photo editing or image segmentation tasks require a dense depth map, with possible relaxation regarding accuracy.

Second, the depth map may include speckle noise, and may represent inaccurate 3D object edges (i.e. *occlusion edges*). The issue is especially noticeable for objects with a fine (i.e. detailed) geometry, for example crowns of trees. It is easy to notice this issue from Figure 1, right. For photo editing, object detection, or image segmentation tasks, clear and geometrically meaningful edges are often more important than accurate depth values at pixels close to those edges.

Third, a depth map may involve invalid step-edges. Stereo matching methods detect *disparities* between base and match images, which are typically measured in integers only (i.e. distances in pixel coordinates). The depth of a point is inversely proportional to its disparity value. Thus, depth does not change smoothly when only using integer disparities. For example, if 2 and 1 are the disparity values of two adjacent pixels then depth varies by factor 2 at those pixels.

We observe that (e.g.) the base image contains some information about scene geometry which is of potential use for enhancing the depth map. Based on this observation, we present a novel depth-refinement method which provides some kind of (e.g. approximate) solutions to the three problems mentioned above. Our method takes as input a base

[1] iSGM was the winning stereo matcher of the “Robust Vision Challenge” at ECCV 2012

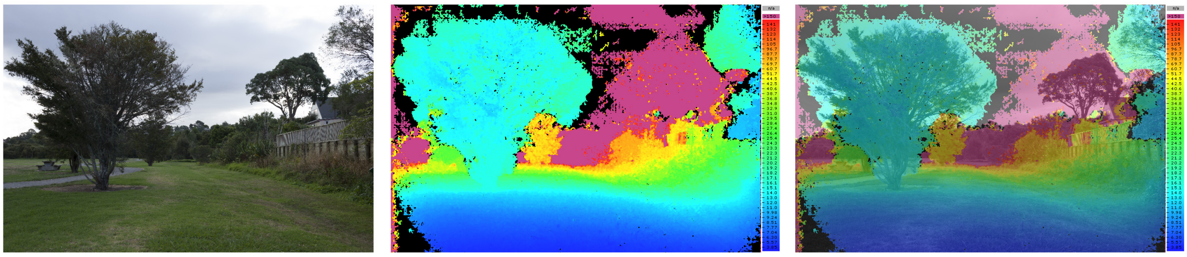

Figure 1: Results of a stereo matcher are imperfect. *Left:* **Base image.** *Middle:* **A disparity map generated by `OpenCV`'s SGBM matcher.** *Right:* **Transparent layers of base image and disparity map illustrate accuracy issues.**

image of a stereo pair, as well as a corresponding disparity map generated by a stereo matcher. We generate an enhanced dense disparity map.

Our method is defined by a four-step process which basically differs from our method applied in [10] (where mean-shift segmentation has been used for disparity map enhancements):

- First, we preprocess the given disparity map by revising disparities in the sky region, and by suppressing incorrect information on the left-hand side of the disparity map (typically caused by occlusion).
- Second, the base image is segmented into several components by spectral matting; we refer to [11] for this technique. The resulting matting components serve as masks in the following refinement process to protect and to enhance object edges.
- Third, we create a disparity layer for each matting component by running a weighted median filter on the pre-filtered disparity map within the segment defined by the component.
- Finally, we calculate the enhanced disparity map by combining those disparity layers. In such a final disparity map, holes are filled, 3D object edges are revised, and speckle noise is reduced.

Enhanced disparity map values are floating point numbers, and not just integers as in the initial map.

The rest of the paper is structured as follows. In Section 2 we recall algorithms as used in our method. Section 3 provides details for our depth-refinement method. Experimental results are shown and discussed in Section 4. Section 5 concludes.

2. BASIC CONCEPTS AND NOTATIONS

This section briefly recalls computer vision algorithms that are used in our approach.

2.1 Stereo Matching

Research on stereo vision has a long history; for example, see [16]. The underlying methodology is close to principles of human vision defined by identifying corresponding points in a pair of images, defining disparities. By applying a triangulation method, depth can be derived from disparities. Two points in base and match image are *corresponding* if they are projections of the same point P in the 3D world. A point P "at infinity" (e.g. very far away such as in the sky) defines disparity 0, and disparities increase if P moves closer to the recording cameras.

A stereo-vision process in high-accuracy stereo-vision applications involves camera calibration (of intrinsic and extrinsic camera parameters), image rectification (for mapping base and match image into a canonical stereo geometry where epipolar lines are identical image rows in both images), stereo matching for identifying corresponding pixels on epipolar lines (thus identifying disparities), confidence evaluation of calculated disparities, and, finally, applying triangulation for mapping disparities into depth. We do not provide any (formal) details here; they can all be found in textbooks such as [6].

Good performing stereo matchers are, for example, based on belief propagation [2], or on *semi-global matching* (SGM) [3, 4]. Stereo matching aims at solving an ill-posed problem, to identify exactly one matching pixel in a match image starting with one pixel in a base image [6]; difficulties for solving this problem arise for many reasons, and we stated three of them above. Algorithms use a smoothness constraint which also causes "blurred" disparities at occlusion edges. For our purpose we like to "sharpen" disparity values at those edges; for this reason we "merge" stereo matching results with spectral matting results in the base image.

2.2 Stereo Refinement

The *bilateral filter* [17], also known as "surface blur", is a selective mean filter for image smoothing or noise reduction. The filter does a weighted average for each pixel p in image I in a window W_p considering both spatial distance and color intensity distance between pixels:

$$I_{bilateral}(p) = \frac{1}{\omega_p} \sum_{p_i \in W_p} I(p_i) \cdot f_c(\|I(p_i) - I(p)\|) \cdot f_s(\|p_i - p\|) \qquad (1)$$

where $\|\ldots\|$ denotes the L_2 norm (here in $\mathbb{R}^2$ or $\mathbb{R}^3$), f_c is the kernel for color-intensity distances between pixels, f_s is the kernel for the spatial distance between pixels, and ω_p is

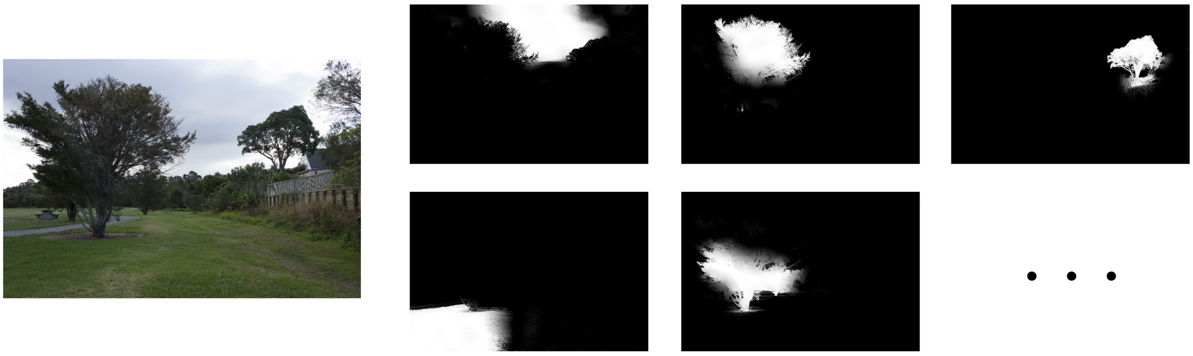

Figure 2: Spectral matting. *Left:* **Input image.** *Right:* **Five matting components $\alpha_k(p)$; the three dots stand for "and so forth".**

a normalization parameter defined by

$$\omega_p = \sum_{p_i \in W_p} f_c(\|I(p_i) - I(p)\|) \cdot f_s(\|p_i - p\|) \qquad (2)$$

Here, f_c and f_s can be Gaussian functions.

The joint bilateral filter, a variation of a bilateral filter, has been developed for depth refinement [7, 12]. This filter uses the color information of the base image I to specify the kernel, and then refines the corresponding disparity map d:

$$\begin{aligned} & d_{J\_bilateral}(p) \\ & = \frac{1}{\omega_p} \sum_{p_i \in W_p} d(p_i) \cdot f_c(\|I(p_i) - I(p)\|) \cdot f_s(\|p_i - p\|) \qquad (3) \end{aligned}$$

The bilateral filter does not discard invalid information. Instead, it spreads inaccurate values to adjacent regions. The joint bilateral filter omits outliers (e.g. inaccurate edges or speckle noise). In comparison, the median filter is more robust on outliers. The *median filter* is a nonlinear operation which runs through an image I and replaces each pixel value $I(p)$ by the median value of neighboring pixels within a $(2m + 1) \times (2m + 1)$ window W_p:

$$I_{median}(p) = \text{median}\{I(p_i) : \; p_i \in W_p\} \qquad (4)$$

By applying a similar idea as the one underlying the joint bilateral filter, the median filter can also be modified for depth refinement (in collaboration with image segmentation) [10]:

$$\begin{aligned} d_{J\_median}(p) \;\; = \;\; & \text{median}\{d(p_i) \neq \text{NA} : \; p_i \in W_p \cap S_p\} \\ & \text{for } S_p \in \mathbf{S}, \quad p \in S_p \qquad (5) \end{aligned}$$

where $\mathbf{S}$ is a family of image segments of I, S_p is that segment which contains p, and NA stands for ”non assigned” (i.e. the value at a pixel location where a disparity was not assigned due to low confidence or occlusion).

2.3 Image Matting

Image matting refers to the problem of accurately extracting foreground objects from an image I [18], defining a set F of foreground pixels, partially overlapping (near borders) with a set B of background pixels. Mathematically, the observed image I is modelled as a linear combination of foreground F and background B by using a *matting parameter* α:

$$I(p) = \alpha(p) \cdot F(p) + (1 - \alpha(p)) \cdot B(p) \qquad (6)$$

with $\alpha(p) \in [0, 1]$.

The matting problem can be generalized by considering multiple matting components $\alpha_k(p)$. Figure 2 illustrates five of such components. Consider K components, specifying K *layers* $F_1, ..., F_K$:

$$I(p) = \sum_{k=1}^{K} \alpha_k(p) F_k(p), \quad \text{for } \alpha_k(p) \in [0, 1], \quad \sum_{k=1}^{K} \alpha_k(p) = 1 \qquad (7)$$

The vectors $[\alpha_1, \ldots, \alpha_K]^\top$ denote the *matting parameters* of the image, which specify the contribution of each layer to the final color observed at each pixel location p.

Image matting can be seen as "soft" image segmentation. Compared to "hard" segmentation [5], image matting can represent object edges more accurately, especially for complex structures which contain detailed sub-pixel information. Typically, image matting methods include user interactions such as the generation of a *trimap* or of some *scribbles* which indicate definite foreground or definite background pixels.

Spectral matting [11], which is used in our method, can automatically segment an input image into several matting components. The method analyzes the smallest (by magnitude) eigenvectors of the image's Laplacian matrix, and obtains the matting components via a linear transformation of those smallest eigenvectors.

3. DEPTH REFINEMENT

Given is a base image I and a corresponding disparity map d. Let Ω be the rectangular $N_{cols} \times N_{rows}$ set of all pixel locations of I.

3.1 Depth Map Pre-Filtering

Sky is typically present in outdoor photos, and the sky region is often large and of little texture. Stereo matching results in such a region often contain mis-matches in large image regions. For outdoor photos, we detect sky regions in I,

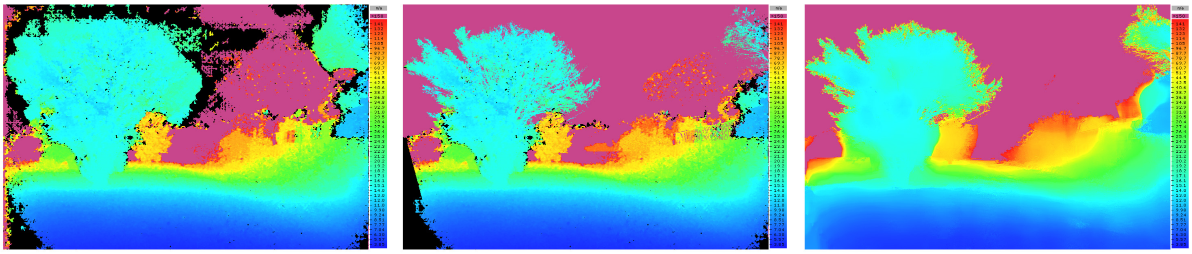

Figure 3: Disparity map refinement. *Left:* **Input disparity map** d. *Middle:* **Pre-filtered disparity map** $\hat{d}$. *Right:* **The final result.**

and set the corresponding values in d uniformly to 0, which marks the sky region as being at infinity.

Blue sky and cloud regions have both high values in the blue channel of I. Thus, we define a pixel p to be a *sky pixel* if p is in the upper half of Ω (i.e. in Ω_{upper}), and its blue channel value $B(p)$ is larger than a threshold T_{sky}:

$$d(p) = 0 \quad \text{if} \quad p \in \Omega_{upper} \wedge B(p) > T_{sky} \tag{8}$$

We use $T_{sky} = 0.8 \cdot G_{max}$ in experiments, where G_{max} is the maximum level in any of the three color channels of I.

Due to the nature of stereo vision, a part of the scene on the left-hand side (i.e. in Ω_{left}) of image I is not included in the match image. Thus, accurate depth information of this region cannot be generated by stereo matching. In general, a usual practice is to discard I and d on Ω_{left}. For the purpose of photo editing, we aim at repairing data in Ω_{left} based on available information. We remove erroneous information in Ω_{left}; resulting gaps are repaired by the process discussed in Section 3.3.

We identify outliers in Ω_{left} by using the calculated disparity map d, since cases of occlusion can be understood in terms of estimated distances to objects. First, we consider a pixel location $p = (x_p, y_p)$ to be in Ω_{left} if

$$x_p < \frac{1}{N_{cols}} \cdot \sum_{x=1}^{N_{cols}} d(x, y_p) + C_{relax} \tag{9}$$

i.e., the mean disparity value in a row specifies how far Ω_{left} extends into this row. We use $C_{relax} = 0.02 \cdot N_{cols}$ in experiments. Next, we identify $d(p)$ as being an outlier if this value deviates too much from the mean disparity in its row. We set $d(p) = \text{NA}$ if $p \in \Omega_{left}$ and

$$\left| d(p) - \frac{1}{N_{cols}} \cdot \sum_{x=1}^{N_{cols}} d(x, y_p) \right| > T_{outlier} \tag{10}$$

We use $T_{outlier} = 0.3 \cdot d_{max}$ in experiments, where d_{max} is the maximum disparity in d. Let $\hat{d}$ be the pre-processed disparity map. See Figure 3, middle, for an example.

3.2 Matting

In order to protect or enhance the occlusion edges of objects in $\hat{d}$, we employ the spectral matting method [11] to segment the input base image I into K layers defined by matting components $\alpha_1, ..., \alpha_K$. Figure 2, right, shows five of those generated matting components for the input image shown on the left. Those components serve as masks in the following refinement process.

The spectral matting method is build on the observation that the smallest eigenvectors of a matting Laplacian L span the individual matting components of the image I. Thus, recovering those matting components is equivalent to finding a linear transformation of the smallest eigenvectors.

Parameters of this method include the number N of smallest eigenvectors which participate in the linear transformation, and the desired number K of the components. Generally, a larger N leads to more accurate results, but also to a higher computational complexity; the number K specifies the balance between under- and over-segmentation. We prefer over-segmentation rather than under-segmentation. In our experiments, we use $N = 400$ and $K = 30$, which lead to "good" results for the considered input images; see Section 4. For details of spectral matting, we refer to [11].

3.3 Weighted Median Filtering

Having K matting components $\alpha_1, ..., \alpha_K$ and the pre-filtered disparity map $\hat{d}$, we create K disparity layers $d_1, ...d_K$ accordingly, and run the median filter in each layer d_k, using α_k as a mask in order to fill holes and to remove noise.

Due to our over-segmentation strategy, normally each of the matting components corresponds to a particular object, or to a part of an object shown in I. Thus, by using α_k as a mask, we can limit the "interference" between different objects, and thus enhance the occlusion edges of objects. The matting components are not always perfect. A component α_k may contain information about adjacent objects at low values due to some color similarity. For reducing this disturbance, we add a weight to the median filter:

$$\begin{aligned} d_k(p) &= \text{WeightedMedian}\{< \hat{d}(p_i), \alpha_k(p_i)^2 >\} \\ \text{where} & \quad \hat{d}(p_i) \neq \text{NA},\ p_i \in W_p,\ \alpha_k(p_i) > 0,\ p \in \Omega, \\ & \quad \text{for } k = 1, ..., K \end{aligned} \tag{11}$$

In the weighted median, we use ${\alpha_k}^2$ as weight, and W_p is again a $(2m+1) \times (2m+1)$ window around p. We use $m = 0.05 \cdot N_{cols}$.

Holes in $\hat{d}$ may go beyond the "handling capacity" of a chosen

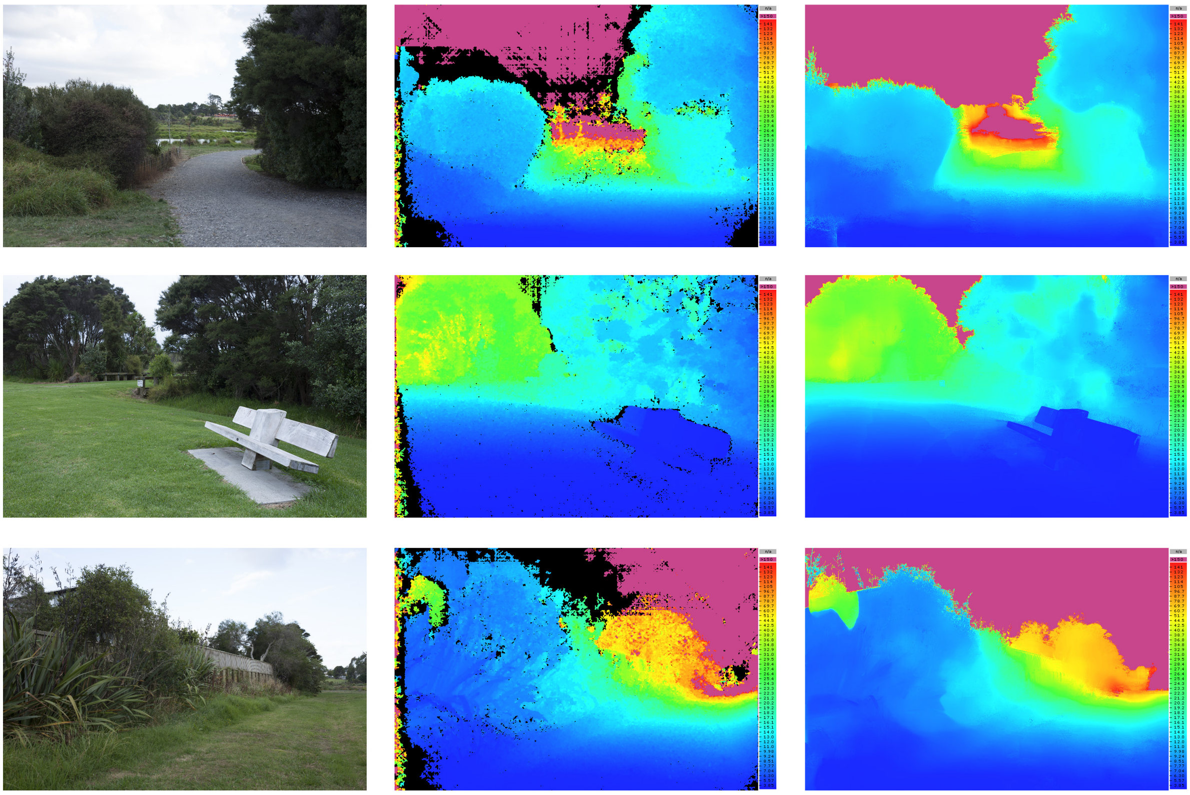

Figure 4: A few results of our method. *Left:* **Base image.** *Middle:* **Original disparity map generated by the SGBM stereo matcher in** `OpenCV`**.** *Right:* **The obtained (i.e. refined) disparity map.**

$(2m+1) \times (2m+1)$ window, i.e. in a case where for pixel location $p \in \Omega$, all the locations in W_p have NA as value. In this case, we run the weighted median filter iteratively on such hole-pixels until all the pixels within the mask have a disparity value assigned.

3.4 Merging of Disparity Layers

Having K refined layers, according to the meaning of the matting components (see Eq. 7), we generate the final disparity map d_{final} by using the weighted mean

$$d_{final}(p) = \sum_{k=1}^{K} \alpha_k(p) d_k(p) \qquad (12)$$

Here, d_{final} is now real-valued instead of integers only. Figure 3, right, shows that the holes in the original disparity map have been filled, and occlusion edges have been corrected.

4. EXPERIMENTS

We illustrate our method for a few outdoor photos, used here with resolution 900 × 600. Figure 4 shows some results. Original disparity maps have been generated by using `OpenCV`'s SGBM stereo matcher. Our experiments show that unavailable depth information (i.e. black pixels in Figure 4) can be filled-in in the refinement process, visually apparent by clearer and more meaningful object edges in the refined depth map.

For verifying the validity of our results in the context of computational photography, we use refined disparity maps d_{final} for darkening the foreground of the base image I. See Fig. 5. The figure illustrates that this depth-aware effect transfers "naturally" into the shown scenes, which indicates the quality of the refined disparity maps.

Figure 6 also shows our result for details in a taken photo. Due to the "soft" character of image matting, disparities change smoothly and naturally from one object to the other, even for the shown complex object shapes.

5. CONCLUSIONS

This paper presents a matting-based stereo refinement process for photo editing. The input data is a base image of a stereo pair and a disparity map generated by a stereo matcher. We pre-filter the given disparity by revising the sky region and by removing incorrect data on the left of the disparity map. Then, the base image is segmented into subtranslucent components using a spectral matting method. For each component, we create a disparity layer. We revise the disparity values in each layer using a weighted median filter. At last, we combine those disparity layers using the subtranslucent matting components, and obtain a real-

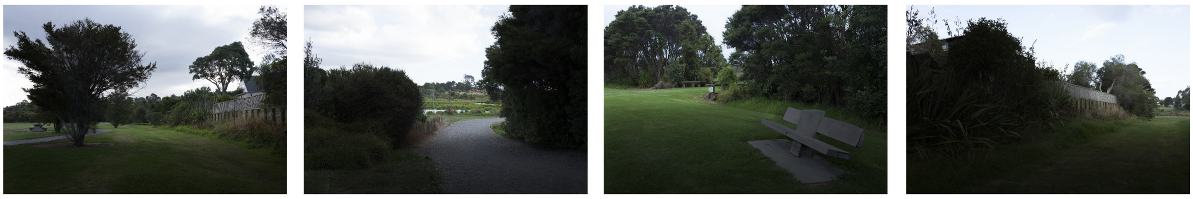

Figure 5: Verification of refined disparity maps by using a simple depth-aware application: darkening of the foreground.

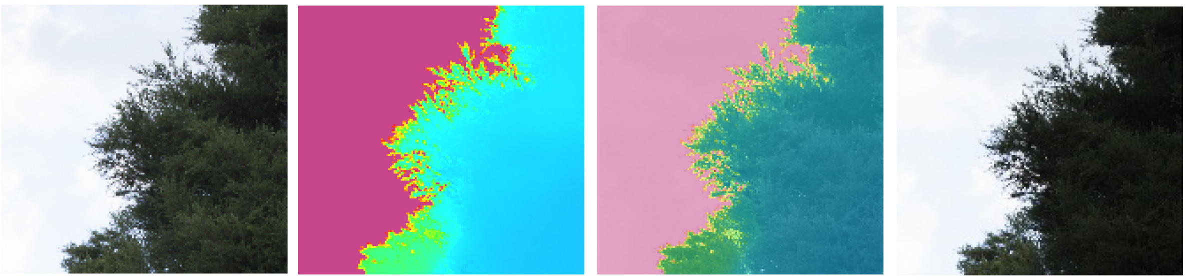

Figure 6: Details of a refined disparity map. *Left:* **Window of the used base image.** *Middle-Left:* **Refined disparity map using our method.** *Middle-Right:* **Overlay of base image and refined disparity map.** *Right:* **Foreground darkening using our refined disparity map.**

valued final disparity map.

Experiments show that our method can revise holes, inaccurate occlusion edges, speckle noise, and invalid step-edges in a given disparity maps. Results are suitable for photo editing, image segmentation, or object detection.

Acknowledgment: This paper is supported by the China Scholarship Council.

6. REFERENCES

[1] Blais, F.: Review of 20 years of range sensor development. J. Electronic Imaging, **13**, 231–240 (2004)

[2] Felzenszwalb, P.F. and Huttenlocher, D.P.: Efficient belief propagation for early vision. *Int. J. Computer Vision*, **70**, 41–54 (2006)

[3] Hirschmüller, H.: Accurate and efficient stereo processing by semi-global matching and mutual information. In Proc. *Computer Vision and Pattern Recognition*, vol. 2, pp. 807–814 (2005)

[4] Hermann, S. and Klette, R.: Iterative semi-global matching for robust driver assistance systems. In Proc. *Asian Conf. Computer Vision*, LNCS, Springer, 465–478 (2013)

[5] Ilea, D.-E. and Whelan, P.-F.: Image segmentation based on the integration of colour–texture descriptors – A review. *Pattern Recognition* **44**, 2479–2501 (2011)

[6] Klette, R.: *Concise Computer Vision - An Introduction into Theory and Algorithms*. Springer, London (2014)

[7] Kopf, J., Cohen, M. F., Lischinski, D., and Uyttendaele, M.: Joint bilateral upsampling. *ACM Tran. Graphics*, **26**, no. 96 (2007)

[8] Khoshelham, K. and Elberink, S.-O.: Accuracy and resolution of Kinect depth data for indoor mapping applications. *Sensors*, **12**, 1437–1454 (2012)

[9] Kopf, J., Neubert, B., Chen, B., Cohen, M., Cohen-Or, D., Deussen, O., Uyttendaele, M., and Lischinski, D.: Deep photo: Model-based photograph enhancement and viewing. ACM Trans. Graphics **27**, No. 116 (2008)

[10] Liu, D. and Klette, R.: Stereo refinement for photo editing. In Proc. *Int. Conf. Computer Vision Graphics*, pp. 391–399 (2014)

[11] Levin, A., Rav Acha, A., Lischinski, D.: Spectral matting. *IEEE Trans. Pattern Analysis Machine Intelligence*, **30**, 1699–1712 (2008)

[12] Matsuo, T., Fukushima, N., and Ishibashi, Y.: Weighted joint bilateral filter with slope depth compensation filter for depth map refinement. In Proc. *Int. Conf. Computer Vision Theory Applications*, (2013)

[13] Ng, R., Levoy, M., Brédif, M., Duval, G., Horowitz, M., and Hanrahan, P.: Light field photography with a hand–held plenoptic camera. *Computer Science Technical Report*, **2**, No. 11 (2005)

[14] Schechner, Y. Y. and Kiryati N.: Depth from defocus vs. stereo: How different really are they?. *Int. J. Computer Vision*, **39**, 141–162 (2000)

[15] Subbarao, M. and Surya, G.: Depth from defocus: a spatial domain approach. *Int. J. Computer Vision*, **13**, 271–294 (1994):

[16] Scharstein, D. and Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *Int. J. Computer Vision*, **47**, 7–42 (2002)

[17] Tomasi, C. and Manduchi, R.: Bilateral filtering for gray and color image. In Proc. *Int. Conf. Computer Vision*, pp. 839–846 (1998)

[18] Wang, J., and Cohen, M.-F.: Image and Video Matting: A Survey. *Foundations Trends Computer Graphics Vision*, **3**, 97–175 (2007)