Real-Time Spatial and Depth Upsampling for Range Data

Xueqin Xiang, Guangxia Li, Jing Tong, Mingmin Zhang, and Zhigeng Pan

State key Lab of Computer Aided Design and Computer Graphics,
Zhejiang University, Hangzhou, 310058, Zhejiang Province, China
{xiangxueqin,lgx,tongjing,zmm,zgpan}@cad.zju.edu.cn

Abstract. Current active 3D range sensors, such as time-of-flight cameras, enable acquiring of range maps at video frame rate. Unfortunately, the resolution of the range maps is quite limited and the captured data are typically contaminated by noise. We therefore present a simple pipeline to enhance the quality as well as improve the spatial and depth resolution of range data in real time. To improve the spatial resolution of range data, we first upsample the depth information with the data from high resolution video camera. And then, a new strategy is utilized to increase the sub-pixel accuracy. We show that these techniques can greatly improve the reconstruction quality, boost the resolution of the range data to that of video sensor while achieving high computational efficiency for a real-time application.

Keywords: Time-of-Flight Camera, Super Resolution, Fast Multi-Lateral Filter, Sub-Pixel Estimation.

1 Introduction

In recent years, a variety of range measuring devices have been developed for 3D data acquisition.For example, by using extremely faster shutter (on the order of nanosecond),time-of-flight(TOF) sensors [1] measure time delay between transmission of a light pulse and detection of the reflected signals on an entire frame at once which are best suited for dynamic scene. This process is largely independent of the scene texture and full frame real-time depth estimates are possible.On the other hand, the main contender to TOF sensor- stereo vision [3] - is rather limited: it is known to be quite fragile in practice (e.g. due to lack of texture).

Unfortunately, being a relatively young technology, TOF sensors have not enjoyed the same advances, with respect to image resolution, quality and photo speed, that have been made in traditional 2D intensity imaging sensors. As a result, in current generation, these sensors provide range data of comparably low image resolution (e.g. only up to 176×144 for MESA SwissRanger^{TM} SR4000 [2]) that are heavily contaminated with noise in the distance measurement.

To overcome this issue, this paper proposes a simple framework to substantially enhance the spatial and depth resolution of range data, e.g., those from the

M.L. Gavrilova et al. (Eds.): Trans. on Comput. Sci. XII, LNCS 6670, pp. 78–97, 2011.

Mesa imaging sensor. To achieve this goal, firstly, we propose a new fast multi-lateral filter, termed as FMLF, to adaptively upsample the low resolution range data in real time by taking advantage of the significant information provided by registered high resolution video camera. To enhance the depth resolution and reduce the discontinuities caused by quantization in the depth map initiation process, a sub-pixel estimation algorithm then is formulated as a Markov Random Field (MRF) and treated it as a Maximum A Posteriori (MAP) problem which can be solved via the gradient descent method.

The main contribution of our method is to present a simple pipeline to enhance the spatial and depth resolution of range data while obtaining real time performance. We also extend our method in a new realm: combined with the low resolution intensity image generated by TOF sensor itself, the quality of range data can be greatly improved.

The rest of the paper is organized as follows. Section 2 introduces the previous works. Section 3 describes the multi-sensor setup of our system. The complete description of the proposed fast and simple super resolution technique is presented in Section 4. The extension is given in Section 5.The experimental results are given in Section 6. Finally the conclusions are outlined in Section 7.

2 Previous Work

There are many approaches that exploit additional information to improve the resolution of range data combining TOF sensor with one or two high resolution video cameras. The main assumption is that depth discontinuities are often related to color changes in the corresponding color image.

Prior researchers often use a probabilistic approach: In [5], MRF is first designed based on the low resolution depth maps and the high resolution camera images and solved via conjugate gradient. Unfortunately, this method gives promising spatial resolution enhancement only up to 10×. Yang et al. [6] then present a method modals a cost volume of depth probability and iteratively applies bilateral filter [7] to refine the cost volume, providing a spatial resolution enhancement of 100× (10× width and 10× height). However, they do not use a joint bilateral filter [8] to link the two images and even with GPU (Graphics Processing Unit) [9] optimization, their effective runtime would be very large due to the number of cost slices and the iterative scheme.

Another recent method [10] utilizes exclusively depth maps, without color image aid: a sequence of low resolution depth maps of same scene is aligned and then merged together to obtain a single depth map with improved resolution. But this method is restricted to static scenes' acquisition. Lindner et al. [23] apply noise and edge aware upsampling for range data. However, using a pure upsampling method, they do not to recover details which are beyond the depth sensor's resolution limit.

Key to our success is the use of multi-lateral filter, which is essential the extension of joint bilateral filter widely used in several state-of-the-art upsampling algorithms [11].Until recently, these edge preserving bilateral filters were too

computationally intensive for real time applications. Several efficient methods [12] enable it to be computed at constant time or even video using GPU implementation [13]. Yang et al. [14] improve on this by not explicitly representing the entire space, but instead sweeping a plane through the intensity level, computing the output in intensity order. This low-memory, cache-friendly algorithm is the fastest known bilateral filter. Inspired by Yang's acceleration strategy, our multi-lateral filter is sliced into one bilateral filter and one joint bilateral filter that computed through discretization technology respectively. Therefore, the real-time performance can be eventually achieved via GPU implementation. What's more, compared with the work of Chan et al. [16], our multi-lateral filter method then allows for sub-pixel accuracy, in contrast with a potentially blocky range result.

TOF sensor also provides an intensity image that is perfectly registered with a depth map at each frame. Since a little interest has been put into this realm [15], we extend our algorithm for improving the quality of range data of TOF sensor by its own low resolution intensity image. Experimental results indicate that even with the help of low resolution intensity image, the quality of range data could be greatly improved by our algorithm.

3 Multi-sensor Setup

We combine a TOF sensor with a high resolution video camera (as shown in Fig.1). In our setup, it has a baseline about 100mm and two sensors are verged towards each other from the parallel setup.

The TOF sensor we have is a SwissRanger SR4000 [2], which can continuously emit a sin wave and detected its reflected signal to produce a depth map in 176 $\times$ 144 size. Its operational range is up to seven meters with the modulation frequency of 20MHZ. In addition, it will also produce an intensity image in the

Fig. 1. Multi-sensor setup

same resolution based on object reflectance. The video camera we have is a dragonfly2 video camera, providing color images with resolution up to 1024 × 768 pixels.

4 Algorithm

An overview of the framework of the approach is given out in Fig.2 and it has two main independent phases: First, up-sample the low resolution range image from TOF sensor to the same size as the high resolution camera image and fast multi-lateral filter (FMLF) is applied for the purpose of spatial super resolution and denoising afterwards. In contrast to Chan's method [16], our fast multi-lateral filter enables of *arbitrary spatial function and arbitrary range function*. To reduce the quantization effect of the depth map (i.e. for the enhancement of depth super resolution), then a sub-pixel refinement algorithm is proposed based on probabilistic model. We will explain the details below.

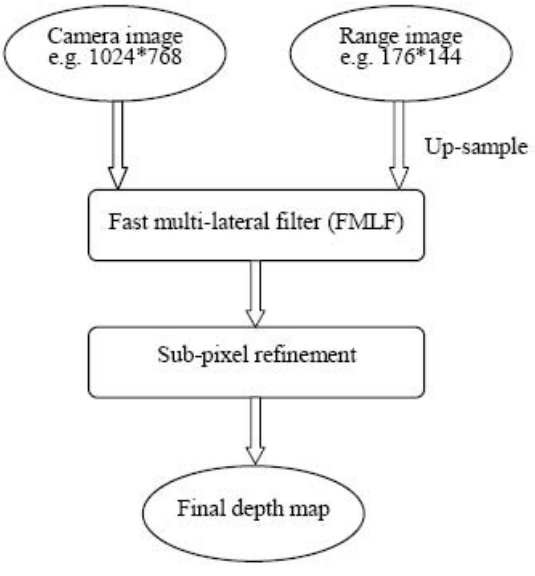


Fig. 2. Pipeline of our algorithm. The range image is up-sampled to the same size as the camera image, and two different images serves as the inputs of the fast multi-bilateral filter. The following is a sub-pixel refinement process.

4.1 FMLF for Depth Upsampling

To cope with the spatial super resolution requirement and meanwhile denosing for noisy real-time 3D sensors, like time-of-flight cameras, we propose a new fast multi-lateral filter for upsampling (FMLF). It is our goal to satisfy spatial super resolution and denosing requirement for real-time 3D sensors as fast as possible

and to make our filter to be more flexible. Like [16], the FMLF filter takes the following form:

$$I_x^F = \frac{1}{K_p} \sum_{y \in N(x)} I(y) f_S(x,y)[(1-\phi(\sigma^2)) f_{\widetilde{R}}(D(x), D(y)) + \phi(\sigma^2) f_R(I(x), I(y))] \quad (1)$$

Where x is a pixel in low resolution range image and y is a pixel in the neighborhood of $N(x)$,$I(x)$ and $I(y)$ are the corresponding range values of pixel x and y, $D(x)$ and $D(y)$ denote the intensity values of pixel x and y in high resolution camera image respectively,f_S, $f_{\widetilde{R}}$ and f_R are all *arbitrary* functions, e.g. Gaussian function or Box function, $\phi(\sigma^2)$ represents a blend function, defining in the interval [0,1], σ^2 is the variance in pixel neighborhood N (e.g. 3×3 lattice) and K_p is a normalizing factor.

From the Equation (1), it is easy to conclude that a low weight ϕ makes our filter behave like a standard joint bilateral filter while a high weight ϕ gives higher influence to the latter range term $f_R(I(x), I(y))$ which makes the filter behave like an edge preserving bilateral filter that smoothes the 3D geometry independently of information from the high resolution camera image.

The main issue is to decide the value of weight ϕ since it controls the characteristic of our filter. We want the filter to switch to a bilateral filter in cases where the areas are actually smooth but heavily contaminated with random noise caused by range measure. Therefore, we intuitively define our blend function $\phi(\sigma^2)$ as follows:

$$\phi(\sigma^2) = \frac{\tau}{\sigma^2 + \tau} \quad (2)$$

Here, σ^2 is the variance in pixel neighborhood in N. We reason that if this variance is large, the local surface patch is most likely to be smooth and only noise-affected - thus former range term $f_{\widetilde{R}}(D(x), D(y))$ should be triggered. Once σ^2 being low, latter range term $f_R(I(x), I(y))$ will be triggered and our filter will act as a bilateral filter to ease the errors caused by range measurement. The unique parameter τ depends on the characteristic of the employed depth sensor and can be determined through experiments.

Please note that the computation of σ^2 on a low-pass filtered depth map is important, because it enables us to reliably disambiguate between isolated random noise peaks and actual depth edges. We have also found that it is better that the range term $f_{\widetilde{R}}$ takes the Gaussian form and cannot simply be set to a box filter in the range domain. With this design, we achieve much better preservation of depth discontinuities if ϕ lies in the transition zone. Finally, by choosing a small spatial support for our FMLF filter (3×3 lattice or 5×5 lattice), any form of texture copy around true depth edges can be reduced in practice while high-quality denoising is still feasible.

4.2 Acceleration Strategy

The complexity of Equation (1) makes direct compute could be time consuming and it is infeasible for real-time application. Several efficient numerical schemes

[14, 17], have been proposed to reduce the computational load of bilateral filter. Inspired by the fastest bilateral filter method [14] so far, our filter is sliced into one bilateral filter and one joint bilateral filter as follow:

$$I_x^F = \frac{1}{K_p} \sum_{y \in N(x)} f_S(x,y)(1-\phi(\sigma^2)) f_{\tilde{R}}(D(x),D(y)) I(y) + \frac{1}{K_p} \sum_{y \in N(x)} f_S(x,y)\phi(\sigma^2) f_R(I(x),I(y)) I(y) \quad (3)$$

Here, the former is a joint bilateral filter while the latter is a bilateral filter. We then could take advantage of acceleration technology proposed by [14]: the range data of low resolution range image and the intensity data of high resolution camera image are discretized into a number of values, compute a linear filter for each such value respectively, the output of which is termed as PBFIC in [14] and get intermediate results by a linear interpolation between two closest PBFICs. The final result is obtained through adding operation between intermediate results. The details are given below:

In practice, we assume that the pixel intensity for an range image $I(x)$ is discrete with $I(x) \in \{0, \cdots, N-1\}$, where N is the total number of grayscale values. Letting $I(x) = k$, the latter term of Equation (3) $\frac{1}{K_p} \sum_{y \in N(x)} f_S(x,y) f_R(I(x), I(y)) I(y)$ can be written as:

$$I^I(x) = \frac{\sum_{y \in N(x)} f_S(x,y) f_R(k, I(y)) I(y)}{\sum_{y \in N(x)} f_S(x,y) f_R(k, I(y))} \quad (4)$$

For every pixel y and every intensity value $k \in \{0, \cdots, N-1\}$, we define:

$$W_k(y) = f_R(k, I(y)) \quad (5)$$

$$J_k(y) = W_k(y) * I(y) \quad (6)$$

Therefore, this bilateral filtering can then be decomposed into N sets of linear filter responses

$$J_k^I(x) = \frac{\sum_{y \in N(x)} f_S(x,y) J_k(y)}{\sum_{y \in N(x)} f_S(x,y) W_k(y)} \quad (7)$$

Thus, we have

$$I^I(x) = J_{I(x)}^I(x) \quad (8)$$

Where J_k^I is defined as Principle Bilateral Filtered Image Component (PBFIC) in [14]. In practice, only $N1$ out N PBFIC ($k \in \{0, \cdots, N1-1\}$) are used. Supposing x is $I(x) \in [L_k, L_{k+1}]$,therefore, the bilateral filtering value $I^I(x)$ can then be linearly interpolated [25] from $J_k^I(x)$ and $J_{k+1}^I(x)$ as following:

$$I^I(x) = (L_{k+1} - I(x)) J_k^I(x) + (I(x) - L_k) J_{k+1}^I(x) \quad (9)$$

Note that, the range filter f_R is not constrained and any desired filter function can be chosen, but approximation can be poor if $N1$ is extremely small for some range filters, e.g., Box filter.

Similarly, the former term of Equation (3) $\frac{1}{K_p}\sum_{y\in N(x)} f_S(x,y) f_{\widetilde{R}}(D(x),D(y))$ $I(y)$ can be reformulated as:

$$I^D(x) = \frac{\sum_{y\in N(x)} f_S(x,y) f_{\widetilde{R}}(k,D(y)) I(y)}{\sum_{y\in N(x)} f_S(x,y) f_{\widetilde{R}}(k,D(y))} \tag{10}$$

Like Equation (9), it can be expressed as:

$$I^D(x) = (P_{k+1} - D(x)) J_k^D(x) + (D(x) - P_k) J_{k+1}^D(x) \tag{11}$$

Where we assume that the intensity of pixel x in high resolution camera image is $D(x) \in [P_k, P_{k+1}]$, and $J_k^D(x)$ is defined according to:

$$J_k^D(x) = \frac{\sum_{y\in N(x)} f_S(x,y) Z_k(y)}{\sum_{y\in N(x)} f_S(x,y) U_k(y)} \tag{12}$$

Where $Z_k(y)$ and $U_k(y)$ are computed as:

$$U_k(y) = f_{\widetilde{R}}(k, D(y)) \tag{13}$$

$$Z_k(y) = U_k(y) * I(y) \tag{14}$$

Finally, we get

$$I_x^F = (1 - \phi(\sigma^2)) I^D(x) + \phi(\sigma^2) I^I(x) \tag{15}$$

These are the two main reasons why our approach outperforms the current state-of-the-art [12] for both accuracy and speed. The main storage required is six memory buffers with the same size as the input image for images. However, [12] requires a set of $N1$ image buffers to store the integral histogram during aggregation. Additionally, in our approach, image pixels are processed independently, allowing for parallel implementation.

Owing to the acceleration strategy discussed above, our GPU implementation of FMLF runs at about 35 frames per second using 8 PBFICs.

4.3 Sub-pixel Estimation

We obtain disparities of the range image on integer level after the process detailed in section above. However, unlike other methods, we also exploit the confidence of an established disparity value.

There has been a growing interest [6, 18] in obtaining accurate sub-pixel disparity since the parabola fitting approaches exhibit artifacts known as pixel-blocking [19]. With the help of Fourier analysis, Scharstein and Szeliski [20] have concluded that sinc interpolator is in theory the best interpolation to evaluate the disparity space image at fractional disparities.

Our approach performs a depth-edge-preserving smoothing on the disparity image, similar to [6] where bilateral filtering was used. Our sub-pixel estimation is similar to adaptive smoothing [26], however, unlike other methods we also further exploit the confidence of an established disparity value.

Our approach treats the sub-pixel estimation as energy minimization problem [27] with:

$$E_{tot} = \sum_{p \in V} E_p(d_p) + \sum_{(p,q) \in D} \alpha E_s(d_p, d_q) \tag{16}$$

Where data term E_p is the cost of assigning disparity d_p to pixel p, pairwise smoothness term E_s is the cost of assigning labels d_p and d_q to two neighboring pixels and α is a scale factor. The higher that α is chosen, the smoother is the resulting disparity map. One could also incorporate the image gradient or the gray value variance as a confidence measure to determine the value of α.

How can an appropriate data term be formulated? Let d_{int} be the integer disparity computed by our fast multi-lateral filter. The data cost of choosing d_p unequal to the former estimated d_{int} is formulated as following:

$$E_p(d_p) = (d_p - d_{int})^2 \tag{17}$$

From the Equation (17), we can conclude that the data cost tends to be small in low-textured regions, whereas it will be large in textured regions.

Let $\widetilde{d}$ be the average disparity within the considered patch D. The smoothness term E_s is defined according to:

$$E_s(d_p, d_q) = (d_p - \widetilde{d})^2 \tag{18}$$

Since our energy Equation (16) has a simple form, it is easy to compute the best solution for a certain point directly instead of to inference by belief propagation (BP) [4, 30] or graph cut (GC) [28, 31]. Partial derivation $\partial E_{tot}/\partial d_p = 0$ yields

$$d_p = \frac{d_{int} + \alpha(N-1)/N * \widetilde{d}}{1 + \alpha(N-1)/N} \approx \frac{d_{int} + \alpha * \widetilde{d}}{1 + \alpha} \tag{19}$$

Where N is the number of pixels within considered patch D. The higher that α is chosen, the smoother is the resulting disparity image.

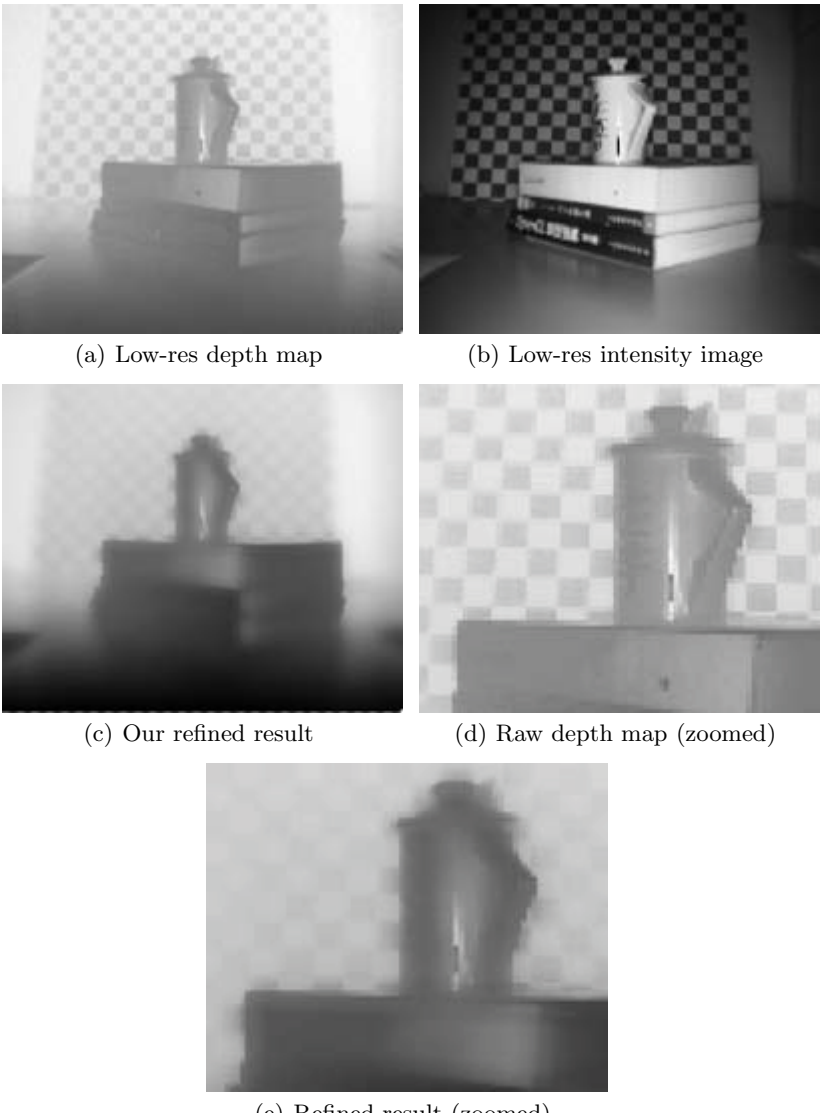
In order to get close to the best solution of the above described problem, we need to iterate Equation (19) to propagate the update disparity values: d_{int} remains the origin value while d_p updated in every iteration.

This sub-pixel estimation favors solutions that are planar in 3D, i.e. fronto-parallel or slanted planes. This way, the algorithm is especially helpful for reconstructing flat object.

See Section 6.3 for results of our sub-pixel estimation.

5 Extended Range Data Super Resolution Based on a Single TOF Sensor

TOF camera robustly provides a range image of real world scenes at video frame rates that is perfectly registered with an intensity image. At this point, it looks like an ordinary color camera plus additional range information. We extend our range data super resolution *only* with a single TOF sensor, based on the



(a) Low-res depth map (b) Low-res intensity image

(c) Our refined result (d) Raw depth map (zoomed)

(e) Refined result (zoomed)

Fig. 3. From a low resolution depth map (a) and a low resolution grayscale intensity map (b) we create a depth map at a higher level quality. The significantly higher quality of our refined result (e) as opposed to the raw depth (d) is obvious.

insight that range measurement may be improved according to the low resolution grayscale intensity image of TOF sensor itself.

Unlike [10], our method relies on one frame and it does not require the setup or calibration process as literature [21] did previously. Therefore, it is available for real-time application, especially within dynamic environment.

Assume the $\widetilde{I}$ denote the low resolution grayscale intensity image obtained from TOF sensor. The fast multi-lateral filter we used is changed into following:

$$I_x^F = \frac{1}{K_p} \sum_{y \in N(x)} I(y) f_S(x,y)[(1-\phi(\sigma^2)) f_{\widetilde{R}}(\widetilde{I}(x), \widetilde{I}(y)) + \phi(\sigma^2) f_R(I(x), I(y))] \quad (20)$$

This is almost identical to Equation (1) with the exceptions that the high resolution camera image is substituted with the low resolution grayscale intensity image.

Equation (20) can be also sliced into one bilateral filter and one joint bilateral filter as follows:

$$I_x^F = \frac{1}{K_p} \sum_{y \in N(x)} f_S(x,y)(1-\phi(\sigma^2)) f_{\widetilde{R}}(\widetilde{I}(x), \widetilde{I}(y)) I(y) + \frac{1}{K_p} \sum_{y \in N(x)} f_S(x,y)\phi(\sigma^2) f_R(I(x), I(y)) I(y) \quad (21)$$

Therefore, the proposed acceleration strategy is utilized for speed up. The subpixel refinement strategy detailed in section 4.3 is also utilized to reduce quantization effect.

At the low integration times required for scene capture at 25 fps, the depth data provided by the SR4000 are severely noise-affected. As shown in Fig.3, our method successfully improves the quality of the low resolution depth maps and the resolution of the depth data can be effectively raised to the level of the video camera. True geometry detailed in data, such as discontinuities, are preserved and enhanced, the random noise level is greatly reduced.

Note that the generic artifacts that arise from the sensitivity of TOF sensor to object reflectance [21] are also prevented. By exploiting the GPU as a fast stream processor, real time performance is feasible. In a word, our design successfully handles the data produced by state-of-the-art time-of-flight sensors which exhibit significantly higher random noise levels than most active scanning devices.

6 Experimental Results

To demonstrate the effectiveness of our approach we applied our technique to various scenes including our own recorded sequences as well as scenes from the Middlebury stereo benchmark datasets [29]. In the following, we discuss implementation details in Section 6.1. Then, we analyze two main aspects of our approach more closely. We first demonstrate the visual superiority of our spatial super resolution results to results obtained with previous upsampling methods, Section 6.2. Thereafter, our depth super resolution results are described in Section 6.3. Finally, we discuss the advantageous run time properties of our algorithm, and discuss practical performance gains in comparison to optimization based upsampling methods, Section 6.4. We end the section by noting some overall limitations from our results and by discussing some possible future directions of investigation for our work, Section 6.5.

6.1 Implementation

Our experimental system consists of a Mesa Swissranger^{TM} SR4000 time-of-flight camera and a Point Gray^{TM} dragonfly2 video camera. The two cameras are placed side-by-side (as closely as possible) and are frame-synchronized. The Swissranger can produce range images with size up to 176×144 pixels and the dragonfly2 can provide color images with resolution up to 1024×768 pixels. To align the range and video images, we resort to the gray-scale intensity images that the Swissranger sensor provides in addition to range images. For the purpose of image registration, the approach proposed in [22] is applied. A better setup would be to use an beam-splitter to align the optical axes of both sensors to guarantee image alignment.

The approach is implemented on a state-of-the-art graphics card. Since our approach operations on individual pixels can be carried out independently, we can capitalize on the massive stream processing power of modern GPUs which by now feature up to 256 individual stream processors. In particular, we employ Nvidia's CUDA programming framework [9] to port our algorithm onto graphics hardware. Overall, we thereby achieve real-time up-sampling and denoising performance.

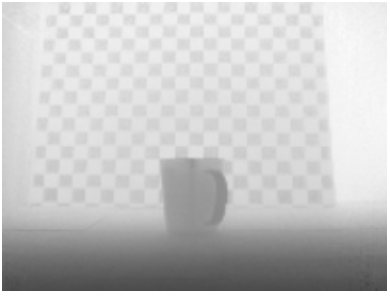
The simplicity of our method lies in that, Compared with previous work [16, 21], only two main parameters are involved in it, they are τ and α. τ is the constant used in Equation (2) which in essential denotes the expected variance due to noise, it is set to 50 experimentally. α is the magnification ratio of smooth term in Equation (16) and is set to 2 in this paper.

6.2 Spatial Super Resolution

Let the f_S in Equation (1) be Box function and $f_{\widetilde{R}}$ and f_R in Equation (1) be all Gaussian functions, we evaluate our algorithms on a real scene where a checkerboard and a cup are involved as shown in Fig.4. It is clear that our method successfully upsamples the low resolution depth maps to high resolution and with respect to the raw 3D data, the visual appearance of depth detail of the checkerboard is improved, especially on textured regions and around boundaries.

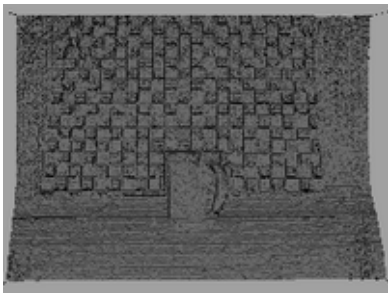
Our approach is also superior to the Chan's method proposed by [16]. Please pay attention to the details, indicated by the red boxes and arrows, for further comparisons. Furthermore, evaluated on Nvidia Geforce 9800 GT platform, the GPU implementation of our approach averages 31ms which is faster than that of the Chan's method (37 ms).

A visual comparison of the depth maps of the Middlebury datasets are provided in Fig.5.The original depth map is down-sample by 8 (2^3) from the ground truth.Currently, f_S is chosen to be Box functions, $f_{\widetilde{R}}$ and f_R in Equation (1) are all chosen to be Gaussian functions. The MRF approach in [5] also improves the stereo quality, but the improvement is relatively small compared to our approach. Cleary, the results using our approach have more clean edges than the input depth maps and the result using MRF approach [5]. According

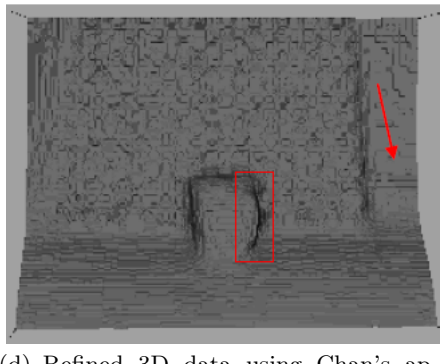


(a) Low-res depth map

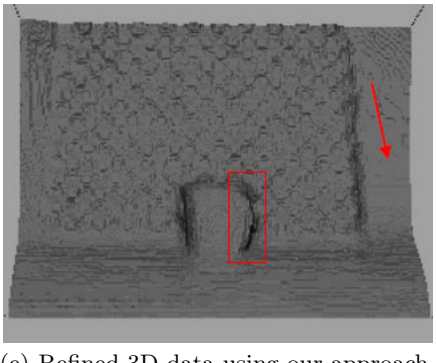
(b) High-res camera image



(c) Raw 3D data



(d) Refined 3D data using Chan's approach



(e) Refined 3D data using our approach

Fig. 4. By using of the high-res camera image (b), our technique upsamples a low-res depth map (a) reconstructed as 3D geometry(c) to a high-res depth map which can be reconstructed as 3D geometry (e) with the comparison of Chan's results (d)

to Fig.5, it is faithfully acknowledged that our results are inferior to the results using Yang's method [6]. However, our approach is designed based on fast and simple pipeline whereas the Yang's method relies on iterations which make it impossible for real-time application.

By visual comparison, our approach outperforms the MRF approach as the resolution of the range sensor keeps on decreasing. In Fig.6, we show that even

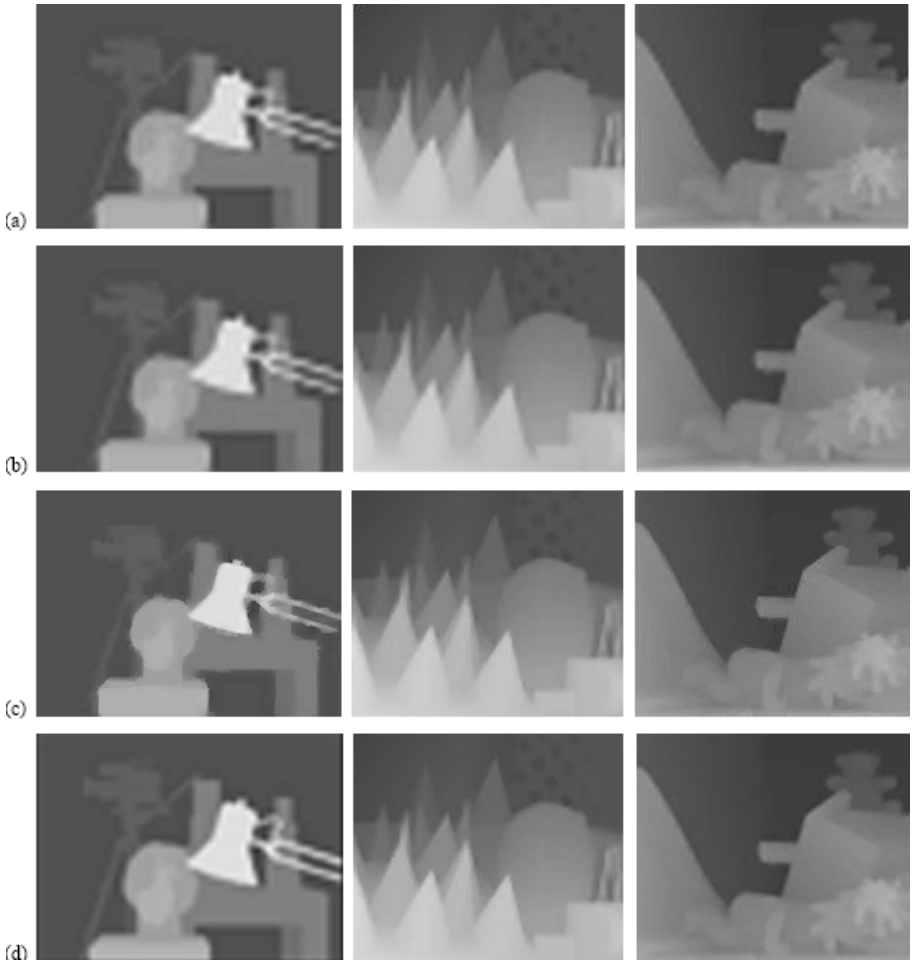
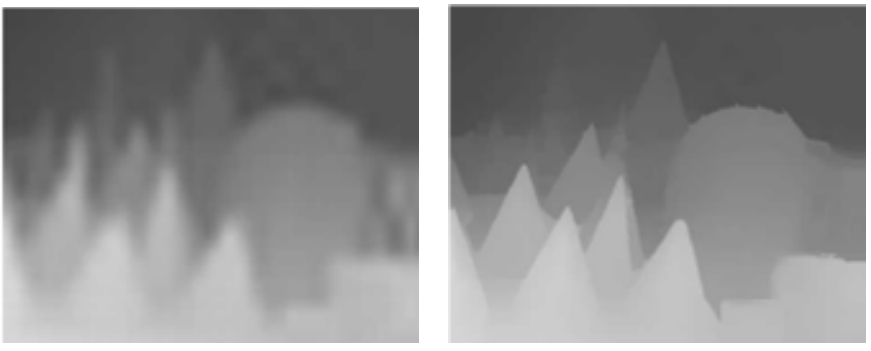


Fig. 5. Super resolution result on Middlebury datasets. (a) Before refinement. (b) Using Diebel's approach [5].(c) Using Yang's approach [6]. (d) Using our approach.

with tiny sensors (down-sample by 16, 2^4), we can still produce decent high-resolution range images.

Fig.7 show the performance of our algorithm on Middlebury datasets when f_S is chosen to be Box function or Gaussian function. Obviously, the two curves (corresponding to f_S is set to be Box function or Gaussian function) are almost coincidence in this experiment. A reasonable explanation may be that test images in Middlebury datasets are well taken under an ideal environment. However, practical experiments have proven that Gaussian function is more robust, especially for noisy cases.



(a) Using Diebel's approach [5] (b) Using our approach

Fig. 6. Super resolution result on Cones dataset (down-sample by 16)

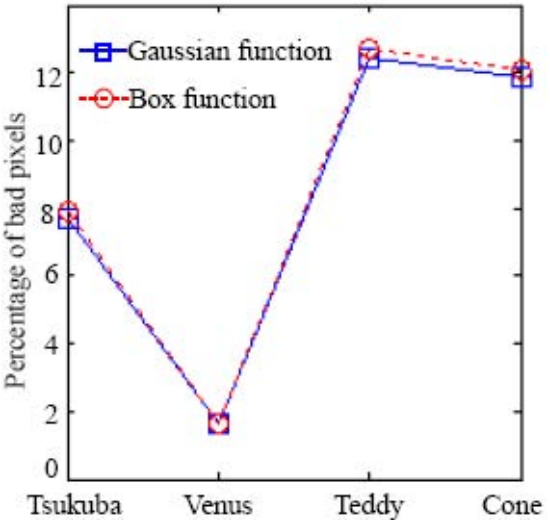
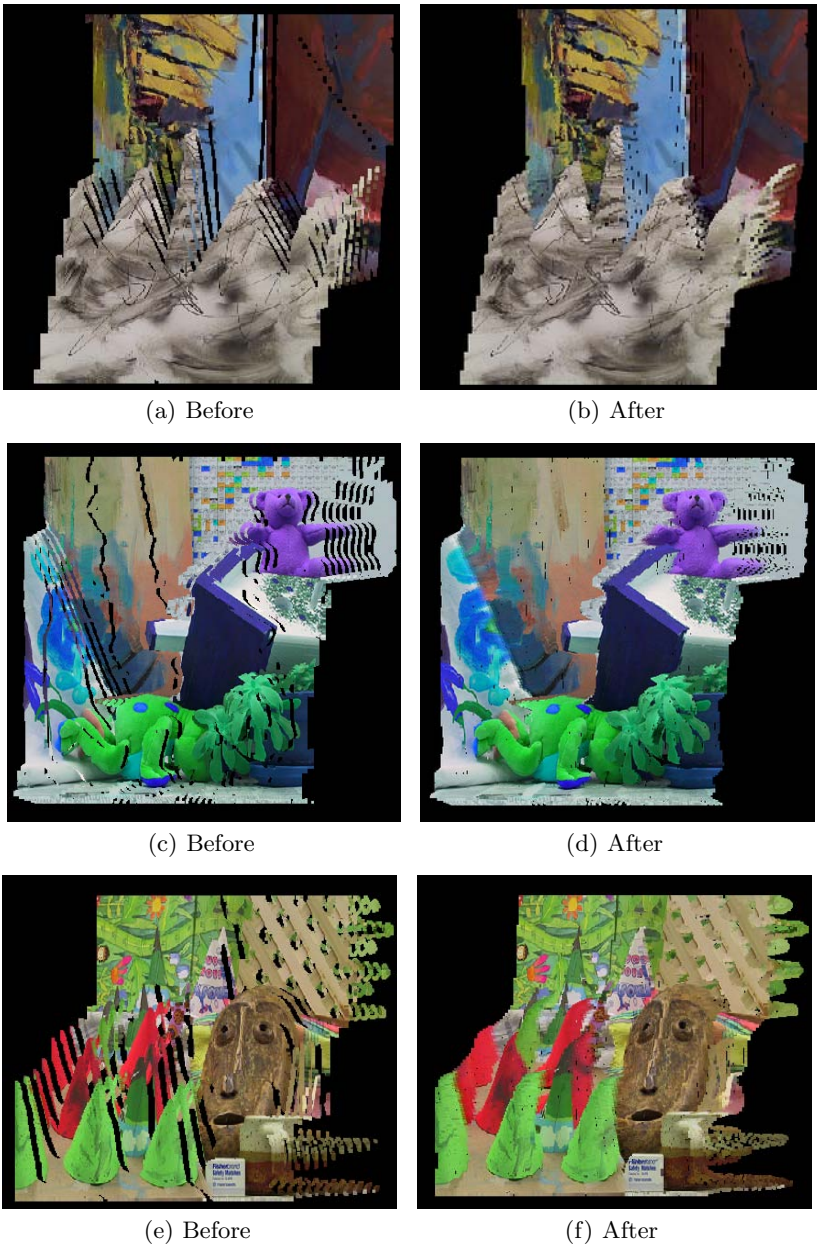


Fig. 7. The performance of our algorithm on Middlebury datasets with regard to f_R being Box or Gaussian function (with error threshold 1)

6.3 Depth Super Resolution

Besides the enhancement of the spatial resolution of range images, our approach also provides sub-pixel estimation for the further enhancement of the depth resolution of range images. A set of synthesized views are shown in Fig.8, providing a visual comparison of the algorithms with and without sub-pixel refinement. The enhancement of the depth resolution is clear: As shown in Fig.8(a), Fig.8(c) and Fig.8(e), the results are quantized into discrete number of planes. After sub-pixel estimation, the quantization effect is removed, as it is shown in Fig.8(b), Fig.8(d) and Fig.8(f).



(a) Before (b) After

(c) Before (d) After

(e) Before (f) After

Fig. 8. Synthesized views produced by our approach before or after sub-pixel estimation

Table 1 evaluates the performance of our approach with and without sub-pixel estimation on Middlebury datasets. The original depth map is down-sample by 8 (2^3) from the ground truth. By comparing bad pixel percentages with

Table 1. Comparison of the results on Middlebury datasets with or without sub-pixel refinement (with error threshold 1, down-sample by 8)

	Tsukuba	*Venus*	*Teddy*	*Cone*
Without sub-pixel refinement	8.23%	1.73%	13.5%	12.9%
With sub-pixel refinement	7.71%	1.62%	12.4%	11.9%

Table 2. Comparison of the results on Middlebury datasets with or without sub-pixel refinement (with error threshold 1, down-sample by 2)

	Tsukuba	*Venus*	*Teddy*	*Cone*
Without sub-pixel refinement	2.45%	0.52%	2.66%	3.25%
With sub-pixel refinement	2.12%	0.43%	2.54%	2.98%

Table 3. Comparison of the results on Middlebury datasets with or without sub-pixel refinement (with error threshold 1, down-sample by 4)

	Tsukuba	*Venus*	*Teddy*	*Cone*
Without sub-pixel refinement	4.93%	1.02%	7.64%	7.42%
With sub-pixel refinement	4.06%	0.58%	6.90%	6.32%

and without sub-pixel estimation, we can conclude that sub-pixel refinement improves the performance of our approach for all data sets.

For further comparison, Table 2 and Table 3 list the performance of our approach with and without sub-pixel estimation on other two scales, i.e., down-sample by 2 (2^1) or 4 (2^2). Clearly, sub-pixel refinement improves the performance of our approach throughout all scales.

6.4 Runtime Analysis

In [5], Diebel et al. use an iterative solver to find the MAP upsampled depth values based on a MRF. Chan et al. [16] report their runtime analysis on Diebel's method: They used an implementation of the error metric and gradient computation on a cpu solved with an iterative L-BFGS-B solver [32] to derive depth results.

In [16], Chan et al. have shown that, to find an optimal set of MRF parameters, it consistently took over 150 iterations to converge to a solution when ran upon upsampling several scenes from the Middlebury dataset. When performing the error metric and gradient computation on the GPU along with appropriate loading of data to and from the GPU, it took on average 755 ms to compute 150 iterations. In the event of a full GPU solution, the iterative runtime of the gradient computation is 279 ms. This value is also absent of the time required for a GPU based solver to find a new gradient direction. In contrast, the GPU

implementation of our approach averages 38 ms, which includes transfering data to and from the GPU.

Our approach is implemented with CUDA technique on a Geforce 9800 GT graphics card (512 MB video memory) GPU, together with a 2.7GHz CPU with dual core. In our experiments, the spatial super resolution phase contributes to major time spent in our algorithm while the run-time of depth super resolution is negligible since the Equation (19) is well suited for parallel execution. Generally speaking, processing an entire video camera image with large size (1024×768 pixels), our approach takes around 40 ms that makes it applausible for real-time applications. The runtime of some experiments is listed in Table4.

Table 4. Runtime on some experiments

	*Figure*3	*Figure*4	*Figure*6
Runtime	19ms	42ms	30ms

Finally, we would like to note that our approach performs much more efficiently than the multi-plane bilateral filtering and upsampling method of Yang et al. [6]. Although their method reportedly produces higher quality results, it will requires many iterations of a bilateral filter at each time step. Therefore, it is infeasible for real-time applications.

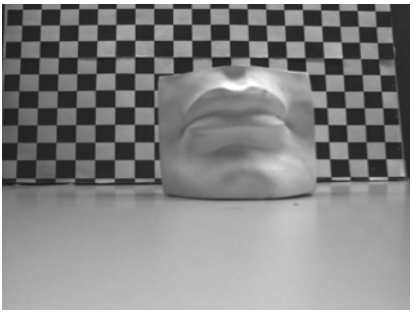
6.5 Discussions and Future Work

Although our approach improves depth details well, it does poorly in some cases. Fig.9 depicts such cases, e.g., the transparent and textureless glass and the high specular head statue. This is because the complimentary nature of the TOF sensor and color camera is invalid. Our formulation (color camera with the TOF sensor) cannot deal with this problem.

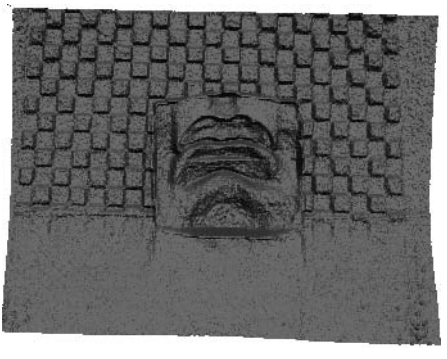
However, 3D shape of specular and transparent objects could be recovered by three viewpoints if incoming light undergoes two reflections or refractions [24]. Although we have not implemented this method, we can image that shape by light path may provide the depth for transparency and textureless objects which is currently not addressed in this paper.

In previous sections we have demonstrated that our approach allows us to produce high-resolution noise-reduced depth data in real-time even with a highly noisy TOF camera. The real-time requirement, however, makes some necessary for which we would like to contrive improved alternatives in the future. First, our FMLF switches between its two operating modes using a fundamentally heuristic model that requires manual parameter setting. In future, we plan to investigate how to learn the correct blending function from real data. Although our approach is defined in a local 3D domain which enables further improvement of our spatial super resolution results, also, we plan to research if it is feasible to improve accuracy of depth super resolution results.

(a) a transparent and textureless glass



(b) a high specular head statue



(c) Raw 3D data from the high specular head statue

Fig. 9. Some problematic cases for our approach

Furthermore, some stepping artifacts in the results are due to 8-PBFICs quantization which we will resolve in future. Finally, we would like to note that our current warping-based alignment, completed before starting experiments, may lead to non-exact depth and video registration in some areas, which may explain remaining blur in our results around some actually sharp depth edges. A feasible hardware solution would have the video and depth sensors record through the same optics which would greatly facilitate alignment.

7 Conclusions

In this paper, we present a fast and simple framework that enable us substantially enhance the spatial and depth resolution of range data in real-time while preserving features, reducing random noise and eliminating artifacts like texture copying phenomenon. We have shown that the results of our approach exceed the reconstruction quality obtainable with related methods from the previous literature. Adapting the fastest acceleration strategy ever known and using the

parallel processing power of a modern graphics processor, the construction of dynamic scene with a high resolution is feasible. In addition, the super resolution method is extended to one single TOF sensor case. Look into future, there are still rooms for improvement. For instance, some constraints and priors (e.g. gradient profile prior) are hoped to be incorporated into our algorithm for further improvement.We also want to investigate how to tackle some difficult cases, such as specular and transparent objects.

Acknowledgements. This work was supported by the National Natural Science Foundation of China (Grant No. 60970076) and the National High Technology Research and Development Program of China (Grant No. 2009AA062704).

References

[1] Ogger, T., Griesbach, K., et al.: 3D-Imaging in real-time with miniaturized optical range camera. In: Proc. OPTO, pp. 89–94 (2004)
[2] SwissRanger^{TM} SR-4000, MESA Imaging inc., `http://www.mesa-imaging.ch`
[3] Yang, Q., Wang, L., Yang, R., Stewnius, H., Nistr, D.: Stereo matching with color-weighted correlation, hierarchical belief propagation, and Occlusion Handling. IEEE Trans. PAMI 31(3), 492–504 (2009)
[4] Liang, C.-K., Cheng, C.-C., Lai, Y.-C., Chen, L.-G., Chen, H.: Hardware efficient belief propagation. In: Proc. CVPR, pp. 80–87 (2009)
[5] Diebel, J., Thrun, S.: An application of markov random fields to range sensing. In: Proc. NIPS (2005)
[6] Yang, Q., Yang, R., Davis, J.: Spatial-depth super resolution for range images. In: Proc. CVPR, pp. 1–8 (2007)
[7] Tomasi, C., Manduchi, R.: Bilateral ltering for gray and color images. In: Proc. ICCV, pp. 839–846 (1998)
[8] Kopf, J., Cohen, M., Lischinski, D., Uyttendaele, M.: Joint bilateral upsampling. ACM Transactions on Graphics (TOG) 26(3), 96(1–5) (2007)
[9] Nvidia Corporation. CUDA: compute unified device architecture programming guide. Technical report (2008)
[10] Schuon, S., Theobalt, C., Davis, J.: Thrun. S.: High-quality scanning using time-of-flight depth superresolution. In: Proc. CVPRW 2008, pp. 1–7 (2008)
[11] Riemens, A.K., Gangwal, O.P., Barenbrug, B., Berretty, R.-P.M.: Multistep joint bilateral depth upsampling. In: SPIE 7257: Proc. VCIP (2009)
[12] Porikli, F.: Constant time O(1) bilateral ltering. In: Proc. CVPR, pp. 1–8 (2008)
[13] Chen, J., Paris, S., Durand, F.: Real-time edge-aware image processing with the bilateral grid. ACM Transactions on Graphics (TOG) 26(3), 103:1–10 (2007)
[14] Yang, Q., Tan, H.-H., Ahuja, N.: Real-time O(1) bilateral ltering. In: Proc. CVPR, pp. 557–564 (2009)
[15] Bhme, M., Haker, M., Martinetz, T., Barth, E.: Shading constraint improves accuracy of time-of-flight measurements. In: Proc. CVPR 2008 (2008)
[16] Chan, D., Buisman, H., Theobalt, C., Thrun, S.: A noise-aware lter for real-time depth upsampling. In: M2SFA 208: Workshop on Multi-camera and Multi-modal Sensor Fusion Algorithms and Applications (2008)
[17] Weiss, B.: Fast median and bilateral ltering. In: Siggraph., vol. 25, pp. 519–526 (2006)

[18] Gehrig, S.K., Franke, U.: Improving stereo sub-pixel accuracy for long range stereo. In: Proc. ICCV, pp. 1–7 (2007)
[19] Shimizu, M., Okutomi, M.: Precise sub-pixel estimation on area-based matching. In: Proc. ICCV, pp. 90–97 (2001)
[20] Szeliski, R., Scharstein, D.: Sampling the disparity space image. IEEE Trans. PAMI 26(3), 419–425 (2004)
[21] Zhu, J., Wang, L., Yang, R., Davis, J.: Fusion of time-of-flight depth and stereo for high accuracy depth maps. In: Proc. CVPR (2008)
[22] Guizar-Sicairos, M., Thurman, S.T., Fienup, J.R.: Efficient subpixel image registration algorithms. Opt. Lett. 33, 156–158 (2008)
[23] Lindner, M., Lambers, M., Kolb, A.: Data fusion and edge-enhanced distance fefinement for 2D RGB and 3D range images. IJISTA, Special Issue on Dynamic 3D Imaging 5(1), 344–354 (2008)
[24] Kutulakos, K.N., Steger, E.: A theory of refractive and specular 3D shape by light-path triangulation. In: Proc. ICCV, pp. 1448–1455 (2005)
[25] Durand, F., Dorsey, J.: Fast bilateral filtering for the display of high-dynamic-range images. In: Siggraph., vol. 21, pp. 1–7 (2002)
[26] Barash, D., Camiciu, D.: A common framework for nonlinear diffusion, adaptive smoothing, bilateral filtering and mean shift. Image and Vision Computing 22(1), 73–81 (2004)
[27] Alvarez, L., Deriche, R., Sanchez, J., Weickert, J.: Dense disparity map estimation respecting image discontinuities: A pde and scale-space based approach. Technical report. Research Report 3874, INRIA Sophia Antipolis, France (January 2000), 2,3
[28] Freeman, W.T., Pasztor, E.C., Carmichael, O.T.: Learning low-level vision. IJCV 40(1), 25–47 (2000)
[29] Middlebury datasets, `http://vision.middlebury.edu/stereo`
[30] Tseng, Y.C., Chang, N., Chang, T.S.: Low memory cost block-based belief propagation for stereo correspondence. In: Proc. ICME, pp. 1415–1418 (2007)
[31] Tappen, M.F., Freeman, W.T.: Comparison of graph cuts with belief propagation for stereo, using identical MRF parameters. In: Proc. ICCV, pp. 900–906 (2003)
[32] Byrd, R., Lu, P., Nocedal, J., Zhu. C.: A limited memory algorithm for bound constrained optimization. SIAM J. Sci. Comp. 16(5), 1190–1208 (1995)