Stereo Matching Techniques for High Dynamic Range Image Pairs

Huei-Yung Lin(✉) and Chung-Chieh Kao

Department of Electrical Engineering, National Chung Cheng University,
Chiayi 621, Taiwan
hylin@ccu.edu.tw

Abstract. We investigate the stereo matching techniques for high dynamic range (HDR) image pairs. It is an emerging topic in computer vision and multimedia applications due to the availability of HDR image capture devices. The disparity computation will eventually take the stereo HDR input. In this work, three state-of-the-art stereo matching algorithms are modified and used to test the advantages of HDR stereo matching. By performing the HDR bit-plane slicing, it is found that only about 16 bits per channel is required for the HDR image format. We propose a 16-bit unsigned integer format to store the HDR image, which allows the available stereo matching algorithms to be adopted for disparity computation. Experiments and performance evaluation are carried out using Middlebury stereo datasets.

Keywords: Stereo matching · High dynamic image

1 Introduction

Stereo matching is a core technology of many 3D related applications. In the multimedia related field, it is commonly used for depth perception, 3D scene reconstruction, depth-image-based rendering, 3DTV and multi-view stereoscopic display, etc. While a large number of algorithms have been developed for stereo correspondence computation in the past few decades [1], it is still difficult to obtain high quality disparity maps for high dynamic range scenes. The main reason is that most current CCD and CMOS sensors are only capable of capturing 2 to 4 orders of light intensity whereas the human eyes are sensitive to around 5 orders of magnitude simultaneously. Consequently, the conventional stereo matching algorithms cannot be successfully performed on the captured images due to the presence of over or under exposed regions.

High dynamic range imaging (HDRI or HDR imaging) serves to represent a real world scene which contains a wide range of luminance change. It adopts the floating point values to encode the large amount of information, instead of using integers as in the conventional image formats. While the existing sensor technology has not caught up to the demands of HDR imaging, a few studios have managed to develop HDR cameras [2]. Their solutions are fairly expensive

T. Bräunl et al. (Eds.): PSIVT 2015, LNCS 9431, pp. 605–616, 2016.
DOI: 10.1007/978-3-319-29451-3_48

and require a long time to capture the full dynamic range of a scene. Therefore, to generate the HDR content in a budget, methods using conventional cameras to take multiple exposures of the same scene to create a single HDR image are commonly adopted.

In computer vision research areas, stereo matching is considered as one of the most challenging and unsolved problems. Since the publication on the taxonomy of stereo algorithms by Scharstein and Szeliski [3], many researchers have participated in an on-line evaluation to compare the performance and accuracy among different stereo matching techniques. Several image datasets with rectified stereo pairs and ground truth disparity maps are available on the Middlebury stereo website as standard test beds [4]. The addition of new and more complex test image datasets provides different scenes taken under 3 illuminations and each with 3 different exposures [5]. It greatly facilitates the stereo matching research on the high dynamic range domain. By constructing HDR images from the given multi-exposure image pairs, the new datasets are suitable for the development and evaluation of HDR stereo matching algorithms.

Among the stereo matching techniques currently available, a popular method for real-time systems is the Semi-Global Block Matching (SemiGlob) algorithm proposed by Hirschmuller [6]. It achieves relatively good quality results while maintaining low computational complexity. This technique is thus successfully employed in mass production vehicles today. Hosni et al. [7] proposed a framework by applying a cost volume filtering method. They have shown that the spatially smooth labeling where the label transitions are aligned with color edges of the image can be efficiently achieved by smoothing the label costs with a fast edge preserving guided filter [8]. Ham et al. [9] describes a steady state matching probability (SSMP) density function to represent the likelihood where the points among the input stereo images being matched. They also focused on using SSMP density function to achieve a probability-based rendering (PBR) method for reconstructing an intermediate view [10]. To develop and investigate the HDR stereo matching techniques, the above algorithms are selected and integrated in our proposed framework.

Although there exist extensive studies on both stereo matching and HDR related fields, not much work has been done on joining these two subjects. Among the similar topics currently under investigation, the most close one is to produce stereoscopic HDR images or videos for high quality 3D contents [11]. However, it is substantially different from stereo matching on HDR images since the disparity computation is not the critical issue. In general, the stereoscopic HDR applications take rough disparity maps and adopt DIBR (depth-image-based rendering) technique to synthesize the stereoscopic image pairs [12]. Stereo matching on HDR images, instead, focuses on how to derive high quality disparity maps in terms of low bad pixel rates and low computation costs.

In recent years, several stereo matching techniques which take HDR image pairs as input have been proposed. Lin et al. [13] present a method to generate high dynamic range and disparity images by simultaneously capturing the high and low exposure images using a pair of cameras. Selmanović et al. [14] propose a technique to generate 3D stereoscopic HDR content using an HDR and LDR pair by stereo correspondence matching. Sun et al. [15] present an algorithm that generates HDR images from multi-exposed LDR stereo images and use a classic NCC (normalized cross-correlation) stereo matching method to evaluate the results. Akhavan et al. [16] discuss the possibilities for combining state-of-the-art stereo matching algorithms with high dynamic range imaging techniques.

In this paper, we present the stereo matching approach using HDR image pairs. The state-of-the-art stereo matching algorithms originally created to run on the conventional (LDR) stereo images are modified for HDR stereo matching. The performance is evaluated on the HDR stereo pairs generated from the multi-exposure images acquired from the same scene. We also adopt the bit-plane slicing techniques originally developed for the LDR image pairs [17], and investigate the feasibility for HDR stereo matching. Experiments are carried out using Middlebury stereo datasets, and three different HDR stereo matching algorithms are evaluated for performance comparison. The results have demonstrated the feasibility of our approach for stereoscopic HDR applications.

2 HDR Images and Stereo Matching

To obtain the HDR images, Reinhard et al. [18] briefly describes how to generate from multiple exposures. By treating the camera response function linearly, we can generate HDR images from multiple LDR exposures by the following steps:

1. Read the different exposure LDR images;
2. Set thresholds to indicate the over and under-exposed pixel intensities of each image and mark them as invalid;
3. Divide LDR images by the exposure time, bringing intensity values of LDR images into a common domain;
4. Accumulate the pixel values that are properly exposed;
5. Normalize the accumulated array by the number of input LDR images that provide valid pixel data for each position.

In the experiments we find setting thresholds to indicate over and under-exposed pixels of each exposure a critical issue if we want to derive good HDR stereo matching results. It is the key for the HDR input to outperform the original LDR input on the stereo matching results.

Since LDR and HDR images are stored in completely different formats [19], it causes many issues to come up when trying to modify LDR stereo matching algorithms for use on HDR stereo image pairs. Most conventional stereo matching algorithms (e.g., SSD, SAD, and SemiGlob) take grayscale image pairs as input. The process of converting RGB images to grayscale is basically joining information from 3 color channels - eliminating hue and saturation information while retaining the luminance.

For HDR images we use the same equation for the conversion of luminance values to a single two-dimensional array. Although the meaning behind applying the same equation to LDR and HDR images is somewhat different, in the experiments we find this method suitable for stereo matching applications. Some stereo matching algorithms in recent years take color image pairs as inputs and use the obtained color information to calculate their matching costs (such as CostFilter and SSMP). For these algorithms we use the color HDR stereo pairs generated from LDR multi-exposed images as input.

While there are various formats for HDR images, we choose the RGBE floating-point encoding format. The pixel values of HDR images are normalized to a range between 0 and 1. This is critical in our research and discussion towards HDR bit-plane slicing. We can then expand and align the whole fraction into an n-bit consecutive memory space. The variable n is determined according to the pixel with the smallest exponent value in the whole image. For example, suppose there is a pixel with the smallest exponent -5 for an HDR image. To allow all fraction bits in the image to be included, n will be $(-1) \cdot ((-5) - 23)$ with 23 being the number of bits in a fraction of a single precision IEEE 754 format floating point number. This allows all fraction bits to be aligned, and provides straightforward bit-plane slicing with fractions.

After the HDR bit-plane slicing process, we can generate bit-level quantized HDR images. For the first bit-level quantized HDR image, all pixels only contain the 2^{-1} bit position data from their original HDR image. For the second bit-level quantized HDR image, all pixels contain 2^{-1} and 2^{-2} data bits from their original HDR images, etc. As a general representation, the k bit-level quantization generate the image given by

$$I(k) = a_{-1} \cdot 2^{-1} + a_{-2} \cdot 2^{-2} + \cdots + a_{-k} \cdot 2^{-k} \tag{1}$$

3 Algorithm and Evaluation

To generate HDR stereo pairs, we use full resolution images (roughly 1300×1100 resolution) from the new stereo datasets available on the Middlebury Stereo Evaluation website. In the years of 2005 and 2006, the datasets provide a total of 30 different scenes. Each scene consists of multiple rectified views taken under three different illuminations (Illum1, Illum2, Illum3) and each with three different exposures (Exp0, Exp1, Exp2). From observation the 3 exposures of each scene all have a consistent ± 2 exposure value range, which allows us to compare the stereo matching results of different scenes and illuminations having the same baseline setup.

By taking multiple exposures, each image in the sequence will have different pixels properly exposed, and other pixels under or over exposed. However, each pixel will be properly exposed in one or more images in the sequence. Under the assumption that the image capture device is perfectly linear, each exposure may be brought into the same domain by dividing each pixel by the image's exposure time. It is therefore possible and desirable to ignore very dark and very bright pixels in the subsequent computations. We set the pixel intensity threshold to mark as over exposed pixels as 250. Any pixel intensity with a value above this number is considered as over exposed and will not be used in the HDR generation process. The under-exposure threshold is set as 5.

After looking into many state-of-the-art stereo matching algorithms from recent years, it is obvious that many algorithm implementations (designed originally for usage on LDR pairs) take advantage on the output-referred LDR image format using 8 bits per color channel. Almost all implementations coded in C store pixel information in an unsigned 8-bit integer space. This not only saves memory usage but also takes advantage of optimized CPU SIMD (single instruction, multiple data) instructions allowing computation on multiple pixel data simultaneously exploiting data level parallelism, and improving array processing performance significantly. To achieve maximum optimization for data parallelism using SIMD instruction sets, often they are hardcoded into programs, meaning it will be a lot of work for them to run on floating point numbers, i.e., HDR images. Another issue is due to the fact that many widely used image libraries such as OpenCV do not support HDR image formats. If one is to perform any kind of research on HDR images, it would be required to spend numerous hours of coding on basic operations that are available in common LDR image libraries.

In this work, we surveyed top performing stereo matching algorithms and found stereo methods CostFilter, SemiGlob, and SSMP suitable for modification to achieve HDR stereo matching. We then focus on the comparison between HDR and LDR stereo matching results. In the experiments, the original CostFilter, SemiGlob and SSMP algorithms are used to run on the 3 exposures of each scene. Modified versions of these three algorithms are used to run on the HDR stereo pairs created from the 3 different exposure images. The scene 'Midd1' as shown in Fig. 1 is an example of the HDR stereo matching outperforming the conventional methods using LDR exposures as input. This is due to the fact that the over and under-exposed pixel intensity values are properly clipped and eliminated in the HDR generation process. The middle exposure (Exp1), which is the correct exposure indicated by the camera, has the best disparity results among the 3 LDR exposures. The under exposure (Exp0) result is a little worse than Exp1. This is because the under-exposed image (Exp0) have darkened regions leaving not enough data to perform accurate stereo matching. A similar situation happens to the over-exposed image pair. By observing the original LDR images of Exposure 2 (Exp2), we can see that the whole image is way too bright, which causes many details and objects disappeared in the image.

Stereo matching requires pixels to have distinct values between each other in order to find the corresponding location of the same object in left and right stereo pairs. If the pixels are over-exposed leaving no texture or detail for stereo matching methods to compare, the disparity map will have a very poor quality result, no matter how robust the algorithm is capable in the case of correctly exposed stereo input. Although performing stereo matching on under- and over-exposed images alone shows disappointing stereo matching results, combining all exposures into an HDR representation can benefit from the extra image data.

Figure 2 shows the results of another experiment using 'Wood1' dataset. In the image of Exposure 0 (Exp0), we can see many dark regions lacking details of the objects in the scene. In the HDR generation process these under-exposed pixels are not eliminated properly, which causes less accurate HDR stereo matching results. When generating the HDR images from multiple exposures, we use the

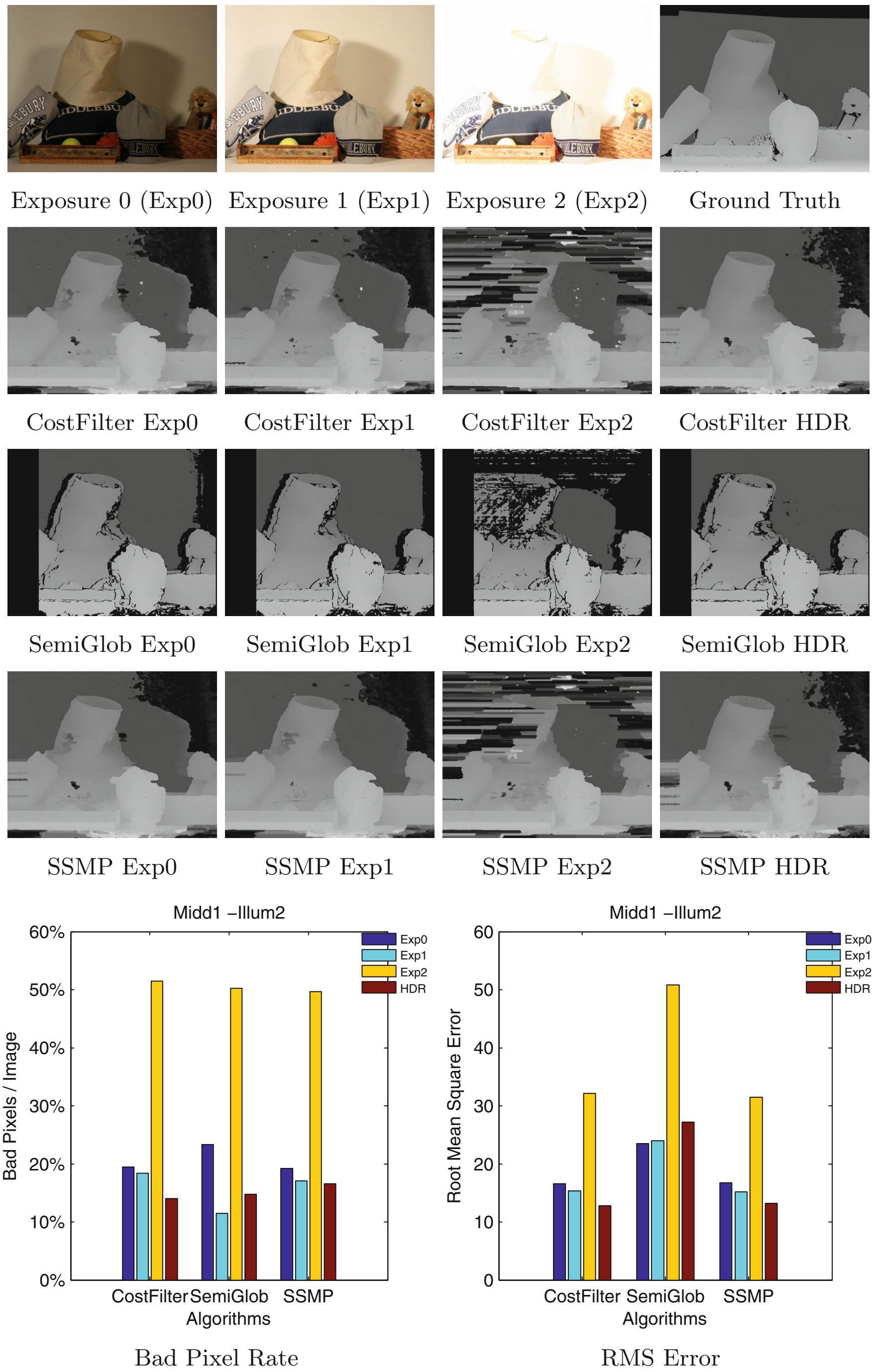

Fig. 1. Scene 'Midd1' Illumination 2 (Illum2) LDR vs HDR results.

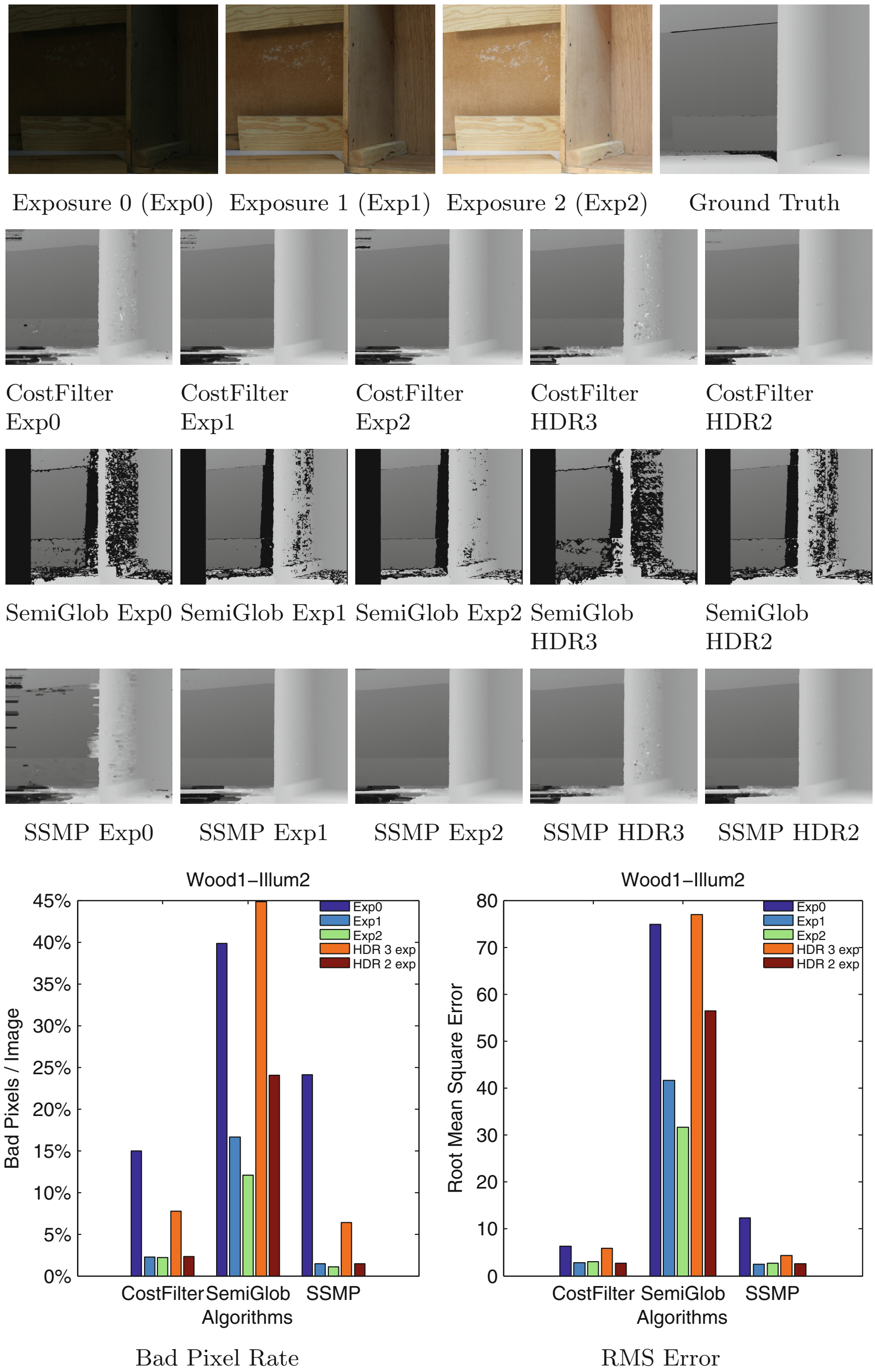

Fig. 2. Scene 'Wood1' Single Illumination (Illum2) LDR vs HDR results. HDR3 indicates created from 3 exposures. HDR2 indicates created from 2 exposures (Exp1, Exp2)

threshold 5 to eliminate pixels that are too dark and 250 to eliminate pixels that are too bright. These parameter settings are clearly not suitable for all scenarios. If we use only Exp1 and Exp2 to generate the HDR images, then the stereo matching results become more acceptable, as illustrated in the figure. Thus, to get the best results from these 3 exposures, we need to find the suitable under- and over-exposure thresholds for clipping pixel values in the HDR generation process. If incorrectly exposed pixels in the images are not eliminated properly, these pixel values will propagate and affect the resulting HDR image, which in turn causes the HDR matching results worse than the correctly exposed LDR ones. We can therefore conclude this experiment with the fact that HDR outperforms LDR only when the intensity clipping thresholds are set properly.

All stereo matching results of the first experiment are tabulated in Table 1. In the 5 scenes tested, HDR versions of 'Midd1', 'Midd2', and 'Monopoly' scenes clearly outperform their corresponding LDR results. This is due to the fact that from the HDR generation process, over-exposed and under-exposed pixel intensity values are properly clipped and eliminated from the HDR generation process. On the other hand, the scenes "Reindeer" and "Wood1" show poor HDR stereo matching results.

Table 1. LDR vs HDR (3 exposures) stereo matching bad pixel rates.

	Aloe Illum1			Aloe Illum2			Aloe Illum3			Baby1 Illum1		
Algorithms	**Exp0**	**Exp1**	**Exp2**	**Exp0**	**Exp1**	**Exp2**	**Exp0**	**Exp1**	**Exp2**	**Exp0**	**Exp1**	**Exp2**
CostFilter-HDR		10.5601			9.485			10.2241			15.8739	
CostFilter-LDR	14.3107	8.6379	***7.3832***	12.1138	7.6327	***6.8886***	12.7334	7.6222	***6.5761***	22.1137	12.61	***11.7812***
SemiGlob-HDR		25.6981			24.5632			25.7534			31.0712	
SemiGlob-LDR	25.0211	19.5558	***18.5555***	23.3308	19.2169	***18.3614***	25.0355	20.468	***19.2431***	29.8626	19.8844	***16.6329***
SSMP-HDR		11.0758			10.6646			10.3838			19.9357	
SSMP-LDR	19.8795	10.3882	***8.8466***	16.289	8.937	***7.6195***	14.7509	9.0516	***7.9415***	38.3126	14.0249	***12.0671***
	Baby1 Illum2			**Baby1 Illum3**			**Cloth3 Illum1**			**Cloth3 Illum2**		
Algorithms	**Exp0**	**Exp1**	**Exp2**	**Exp0**	**Exp1**	**Exp2**	**Exp0**	**Exp1**	**Exp2**	**Exp0**	**Exp1**	**Exp2**
CostFilter-HDR		12.8852			12.2907			3.3194			3.1678	
CostFilter-LDR	18.5774	11.1543	***8.7538***	16.4616	10.3675	***10.2523***	4.6538	2.9648	***2.8294***	3.0355	2.8052	***3.1125***
SemiGlob-HDR		28.4723			27.9715			11.2368			7.7316	
SemiGlob-LDR	26.6789	17.4644	***15.4227***	25.9842	16.9775	***14.9178***	11.4334	7.7863	***7.0369***	7.5254	***6.9861***	7.2694
SSMP-HDR		13.6788			12.6247			3.3125			3.4216	
SSMP-LDR	23.3062	11.1051	***8.9452***	21.1075	10.4442	***10.1045***	7.1509	3.0636	***2.9117***	2.8942	***2.8062***	2.9698
	Cloth3 Illum3			**Midd1 Illum2**			**Midd2 Illum2**			**Monopoly Illum2**		
Algorithms	**Exp0**	**Exp1**	**Exp2**	**Exp0**	**Exp1**	**Exp2**	**Exp0**	**Exp1**	**Exp2**	**Exp0**	**Exp1**	**Exp2**
CostFilter-HDR		2.421232			***12.809572***			***9.509858***			***6.370229***	
CostFilter-LDR	***2.318061***	2.253166	2.425352	16.562021	15.32695	32.163753	12.233618	11.80046	30.817319	9.712566	18.539122	32.779233
SemiGlob-HDR		23.417206			27.242021			26.40699			31.439887	
SemiGlob-LDR	***23.208917***	23.550864	24.035603	***23.503187***	23.968582	50.849491	23.905366	***23.052968***	49.761127	***23.110402***	24.31458	50.977137
SSMP-HDR		2.932481			***13.248041***			***10.891008***			***5.132793***	
SSMP-LDR	***2.440035***	2.442828	2.598794	16.71691	15.179717	31.498245	12.169497	11.711232	28.570909	6.827182	15.108296	32.043236
	Reindeer Illum1			**Rocks1 Illum1**			**Rocks1 Illum2**			**Rocks1 Illum3**		
Algorithms	**Exp0**	**Exp1**	**Exp2**	**Exp0**	**Exp1**	**Exp2**	**Exp0**	**Exp1**	**Exp2**	**Exp0**	**Exp1**	**Exp2**
CostFilter-HDR		11.6935			5.9373			5.6882			5.7211	
CostFilter-LDR	14.0529	***10.3011***	11.8272	16.8082	8.094	***5.7901***	11.349	5.3911	***4.3507***	8.5694	5.2482	***4.6236***
SemiGlob-HDR		28.6551			19.0521			17.846			15.4912	
SemiGlob-LDR	27.619	21.7317	***21.5128***	18.6442	9.8143	***7.4289***	16.7113	8.9711	***7.6411***	14.1115	8.9657	***7.7281***
SSMP-HDR		11.79			***4.8183***			4.8455			5.1165	
SSMP-LDR	14.5659	***9.9756***	10.8423	34.9485	7.7635	7.3271	14.6579	4.4092	***4.0265***	8.066	5.0848	***4.6891***
	Rocks2 Illum1			**Rocks2 Illum2**			**Rocks2 Illum3**			**Wood1 Illum2**		
Algorithms	**Exp0**	**Exp1**	**Exp2**	**Exp0**	**Exp1**	**Exp2**	**Exp0**	**Exp1**	**Exp2**	**Exp0**	**Exp1**	**Exp2**
CostFilter-HDR		3.0185			3.1024			2.8792			7.7188	
CostFilter-LDR	6.2622	2.7277	***2.0433***	7.0107	2.8744	***2.0678***	4.3836	2.4363	***2.0873***	14.9723	2.2695	***2.1733***
SemiGlob-HDR		10.9581			12.6551			8.5396			44.875	
SemiGlob-LDR	10.6543	5.5585	***4.1695***	13.6324	5.4548	***3.9418***	7.8741	4.7667	***3.8864***	39.8532	16.6652	***12.0399***
SSMP-HDR		2.5498			2.5044			2.8934			6.392	
SSMP-LDR	7.5783	2.5392	***1.9962***	16.4043	2.7289	***2.2068***	3.9263	2.6156	***2.1688***	24.1315	1.4503	***1.0554***

Table 2. The maximum number of bits required to store the pixel radiance values for different scenes.

Scene	Aloe	Baby1	Cloth3	Rocks1	Rocks2	Midd1	Midd2	Monopoly	Reindeer	Wood1
Illum1	14	15	14	17	14	14		13	14	
Illum2	14	14	14	17	14		15			14
Illum3	14	14	14	16	14					

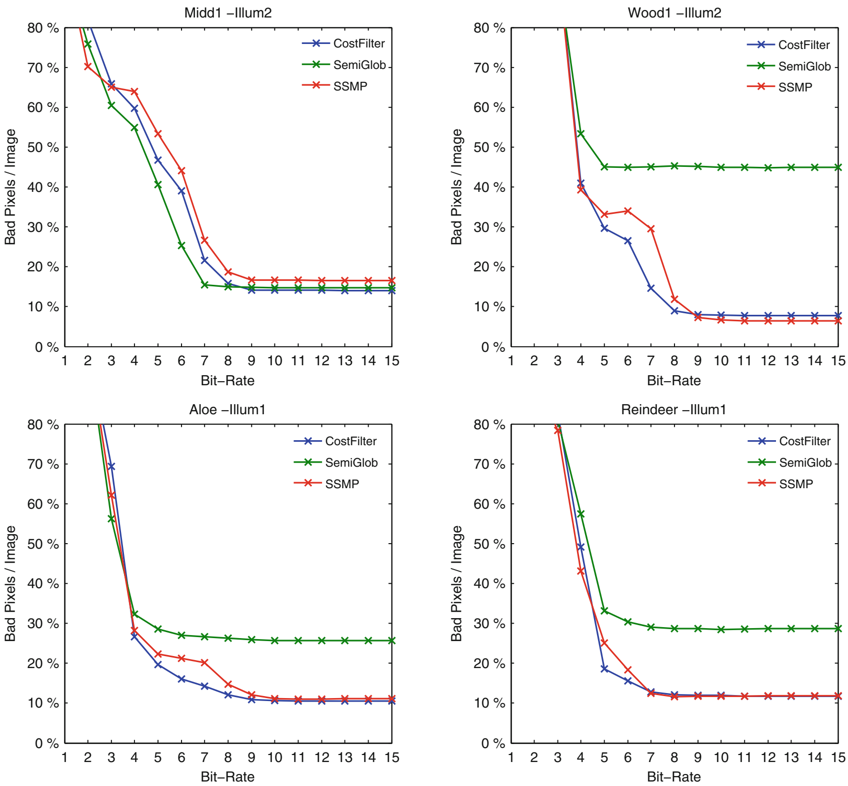

Fig. 3. Some examples of bit-level quantized HDR stereo matching results ('Midd - Illum2', 'Wood1 - Illum2', 'Aloe - Illum1', 'Reindeer - Illum1').

In the second part of experiment and evaluation, we apply the bit-plane slicing technique to generate the bit-level quantized versions of HDR stereo images used in the experiments. Table 2 shows a summary of the maximum required bits to store the radiance information for a pixel in different scene images. The numbers in the table indicate the maximum required value for the variable k given in Eq. (1). In the 'Aloe' scene, for example, all 3 illumination conditions only require 14 bits to store the radiance information. Figure 3 shows some of the bit-level quantization stereo matching results, 'Midd1-Illum2', 'Wood1-Illum2', 'Aloe-Ilum1', and 'Reindeer-Illum1'. The x-axis indicates the value for

Table 3. Comparison of Integer-HDR, HDR and LDR on the bad pixel rate using the algorithm "SemiGlob".

Used Stereo Pairs	Aloe-Illum1			Aloe-Illum2			Aloe-Illum3			Baby1-Illum1		
Exposure	Exp0	Exp1	Exp2	Exp0	Exp1	Exp2	Exp0	Exp1	Exp2	Exp0	Exp1	Exp2
SemiGlob-int16HDR	25.6976			24.576			25.7521			31.0602		
SemiGlob-HDR	25.6981			24.5632			25.7534			31.0712		
SemiGlob-LDR	25.0211	19.5558	18.5555	23.3308	19.2169	18.3614	25.0355	20.468	19.2431	29.8626	19.8844	16.6329
Used Stereo Pairs	**Baby1-Illum2**			**Baby1-Illum3**			**Cloth3-Illum1**			**Cloth3-Illum2**		
Exposure	Exp0	Exp1	Exp2	Exp0	Exp1	Exp2	Exp0	Exp1	Exp2	Exp0	Exp1	Exp2
SemiGlob-int16HDR	28.4684			28.0146			11.6647			8.1422		
SemiGlob-HDR	28.4723			27.9715			11.2368			7.7316		
SemiGlob-LDR	26.6789	17.4644	15.4227	25.9842	16.9775	14.9178	11.4334	7.7863	7.0369	7.5254	6.9861	7.2694
Used Stereo Pairs	**Cloth3-Illum3**			**Midd1-Illum2**			**Midd2-Illum2**			**Monopoly-Illum2**		
Exposure	Exp0	Exp1	Exp2	Exp0	Exp1	Exp2	Exp0	Exp1	Exp2	Exp0	Exp1	Exp2
SemiGlob-int16HDR	8.2729			14.7911			14.0629			17.7825		
SemiGlob-HDR	7.8652			14.7794			14.059			17.8097		
SemiGlob-LDR	7.6022	7.0145	7.372	23.3308	11.49	50.2709	12.6123	10.7588	57.6306	14.8002	14.1561	39.4134
Used Stereo Pairs	**Reindeer-Illum1**			**Rocks1-Illum1**			**Rocks1-Illum2**			**Rocks1-Illum3**		
Exposure	Exp0	Exp1	Exp2	Exp0	Exp1	Exp2	Exp0	Exp1	Exp2	Exp0	Exp1	Exp2
SemiGlob-int16HDR	28.6534			19.0506			17.8385			15.4882		
SemiGlob-HDR	28.6551			19.0521			17.846			15.4912		
SemiGlob-LDR	27.619	21.7317	21.5128	18.6442	9.8143	7.4289	16.7113	8.9711	7.6411	14.1115	8.9657	7.7281
Used Stereo Pairs	**Rocks2-Illum1**			**Rocks2-Illum2**			**Rocks2-Illum3**			**Wood1-Illum2**		
Used Stereo Pairs	Exp0	Exp1	Exp2	Exp0	Exp1	Exp2	Exp0	Exp1	Exp2	Exp0	Exp1	Exp2
SemiGlob-int16HDR	10.9569			12.6579			8.5416			45.1112		
SemiGlob-HDR	10.9581			12.6551			8.5396			44.875		
SemiGlob-LDR	10.6543	5.5585	4.1695	13.6324	5.4548	3.9418	7.8741	4.7667	3.8864	39.8532	16.6652	12.0399

the variable k in Eq. (1) while the y-axis indicates the bad pixel rate (in percentage). It is surprised that most scenes need only 70 % of the total bits to achieve the results of the same stereo matching quality as using the full HDR image. For example, 'Midd1-Illum2' only requires 8 bits while the HDR image uses 15 bits to store all HDR radiance values.

In the experiments we find that, although the floating point is used to store the HDR's pixel radiance values, all of the scenes (test images) actually need only a maximum of 17 bits to store the data of each pixel. Furthermore, the stereo matching results as shown in Fig. 3 use only about 10 bits at most to achieve the same quality of disparity results as the full HDR images can provide. This brings up an interesting question - do we really need to store the HDR image data in floating point representation formats? If we are able to store the HDR image radiance values as the widely used LDR file formats (integers), then the research towards HDR stereo matching could greatly benefit. Many graphics libraries and state-of-the-art stereo matching methods are already implemented using integers. There is no need to spend numerous efforts on coding or modifying the available stereo matching codes, but focus on enhancing the stereo correspondence methods instead.

To validate the possibility of integer-HDR stereo matching, we convert the HDR images to an 'integer' format, PNG (Portable Network Graphics), for the last part of experiment and evaluation. PNG is a lossless compression format for LDR bitmap images and is the most used lossless image compression format on the internet. In PNG specification there is a 'Truecolor' option capable of storing 16 bits for each channel pixel. Thus, we can store all HDR radiance values in the

PNG file format by shifting the 16 positions of the floating point to the right and store only the resulting integer part of the number while truncating the rest of the bits. It means that the normalized HDR images with the floating point values in the range of $0 - 1$ becomes integers with the range of $0 - 65535$. We can then read the image data from the stored 'integer-HDR' PNG file and apply the conventional stereo matching algorithms.

Table 3 shows the comparison of bad pixel rates among the disparity maps obtained using PNG-stored 16-bit HDR format (int16-HDR), floating point HDR format and conventional LDR. It shows that the stereo matching quality of 'int16-HDR' is not lost too much, compared to the disparity derived from the original HDR images. This observation is consistent with the HDR encoding research presented by Mantiuk et al. [20]. The HDR pixel values can be represented using as few as $10 - 12$ bits for luminance and 8 bits for chrominance without introducing any visible quantization artifacts. Although the proposed technique is mainly used for HDR video encoding, it still proves it still provides a way to represent HDR images with less bits.

4 Conclusion

In this work, the stereo matching techniques for high dynamic range image pairs are investigated. We address the key issues on generating HDR images suitable for achieving high-quality disparity results. Three state-of-the-art stereo matching algorithms are modified and used to test the performance of our HDR stereo technique. From the analysis of HDR bit-plane slicing, it is found that only about 16 bits per channel are required to store HDR images. Moreover, the results show that using 10 bits of image data can achieve the same disparity quality as the full HDR image pair can provide. We propose a 16-bit unsigned integer format to store the HDR image, which allows the available stereo matching algorithms to be adopted for stereo HDR with slight modification. The experiments have demonstrated that the HDR stereo pairs generated with proper thresholds can provide better disparity results compared to the LDR countparts.

References

1. Dhond, U., Aggarwal, J.: Structure from stereo: a review. IEEE Trans. Syst. Man Cybern. **19**(6), 1489–1510 (1989)
2. Sbaiz, L., Yang, F., Charbon, E., Susstrunk, S., Vetterli, M.: The gigavision camera. In: Proceedings of the 2009 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2009. IEEE Computer Society, Washington, DC, pp. 1093–1096 (2009)
3. Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. Int. J. Comput. Vis. **47**(1–3), 7–42 (2002)
4. Scharstein, D., Szeliski, R.: Middlebury stereo vision page (2002). http://vision.middlebury.edu/stereo

5. Scharstein, D., Pal, C.: Learning conditional random fields for stereo. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2007, pp. 1–8 (2007)
6. Hirschmuller, H.: Stereo processing by semiglobal matching and mutual information. IEEE Trans. Pattern Anal. Mach. Intell. **30**(2), 328–341 (2008)
7. Hosni, A., Rhemann, C., Bleyer, M., Rother, C., Gelautz, M.: Fast cost-volume filtering for visual correspondence and beyond. IEEE Trans. Pattern Anal. Mach. Intell. **35**(2), 504–511 (2013)
8. He, K., Sun, J., Tang, X.: Guided image filtering. In: Daniilidis, Kostas, Maragos, Petros, Paragios, Nikos (eds.) ECCV 2010, Part I. LNCS, vol. 6311, pp. 1–14. Springer, Heidelberg (2010)
9. Ham, B., Min, D., Oh, C., Do, M., Sohn, K.: Probability-based rendering for view synthesis. IEEE Trans. Image Process. **23**(2), 870–884 (2014)
10. Oh, C., Ham, B., Sohn, K.: Probabilistic correspondence matching using random walk with restart. In: Bowden, R., Collomosse, J.P., Mikolajczyk, K., eds.: BMVC, BMVA Press pp. 1–10 (2012)
11. Mann, S., Lo, R., Huang, J., Rampersad, V., Janzen, R.: Hdrchitecture: Real-time stereoscopic hdr imaging for extreme dynamic range. In: ACM SIGGRAPH 2012 Emerging Technologies, SIGGRAPH 2012. ACM, New York, pp. 11:1–11:1 (2012)
12. Zhang, L., Tam, W.: Stereoscopic image generation based on depth images for 3d tv. IEEE Trans. Broadcast. **51**(2), 191–199 (2005)
13. Lin, H.Y., Chang, W.Z.: High dynamic range imaging for stereoscopic scene representation. In: Proceedings of the 16th IEEE International Conference on Image Processing, ICIP 2009. IEEE Press, Piscataway, pp. 4249–4252 (2009)
14. Selmanović, E., Debattista, K., Bashford-Rogers, T., Chalmers, A.: Generating stereoscopic hdr images using hdr-ldr image pairs. ACM Trans. Appl. Percept **10**(1), 3:1–3:18 (2013)
15. Sun, N., Mansour, H., Ward, R.: Hdr image construction from multi-exposed stereo ldr images. In: 2010 17th IEEE International Conference on Image Processing (ICIP), pp. 2973–2976 (2010)
16. Akhavan, T., Yoo, H., Gelautz, M.: A framework for hdr stereo matching using multi-exposed images. In: Proceedings of HDRi2013 - First International Conference and SME Workshop on HDR imaging (2013)
17. Lin, H.Y., Lin, P.Z.: Hierarchical stereo matching with image bit-plane slicing. Mach. Vis. Appl. **24**(5), 883–898 (2013)
18. Reinhard, E., Ward, G., Pattanaik, S., Debevec, P.: High Dynamic Range Imaging: Acquisition, Display, and Image-Based Lighting (The Morgan Kaufmann Series in Computer Graphics). Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (2005)
19. Larson, G.W., Shakespeare, R.: Rendering with Radiance: The Art and Science of Lighting Visualization. Booksurge Llc, New York (2004)
20. Mantiuk, R., Krawczyk, G., Myszkowski, K., Seidel, H.P.: Perception-motivated high dynamic range video encoding. In: ACM SIGGRAPH 2004 Papers, SIGGRAPH 2004. ACM, New York, pp. 733–741 (2004)