World Academy of Science, Engineering and Technology
International Journal of Computer and Systems Engineering
Vol:16, No:5, 2022

Block-Based 2D to 3D Image Conversion Method

S. Sowmyayani, V. Murugan

***Abstract*—**With the advent of three-dimension (3D) technology, there are lots of research in converting 2D images to 3D images. The main difference between 2D and 3D is the visual illusion of depth in 3D images. In the recent era, there are more depth estimation techniques. The objective of this paper is to convert 2D images to 3D images with less computation time. For this, the input image is divided into blocks from which the depth information is obtained. Having the depth information, a depth map is generated. Then the 3D image is warped using the original image and the depth map. The proposed method is tested on Make3D dataset and NYU-V2 dataset. The experimental results are compared with other recent methods. The proposed method proved to work with less computation time and good accuracy.

***Keywords*—**Depth map, 3D image warping, image rendering, bilateral filter, minimum spanning tree.

I. Introduction

THE conversion of 2D images to 3D images is becoming important research with the advent of 3D technology. Conversion of 2D to 3D images plays a crucial role in accomplishing the growth of stereoscopic images of high quality. The most important step of such conversion is depth map generation for the 2D image. A person normally observes the heuristic depth cues for depth perception generation. The key definitions of depth perception are two-eye binocular depth cues and one-eye monocular depth cues [1].

Several algorithms for the generation of depth maps have been developed in recent years according to the theory of the human visual system. A growing algorithm has their pros and cons. Most algorithms for depth estimation use a single depth cue but few use hybrid depth cues to generate depth maps. Mao and Ibsiyasu have developed an algorithm for covering gray-scale images in 2D to 3D. The application of macro-auto-radiography images of rat brains to color-coding showed the advantages of the approach [2].

Chin et al. have implemented 2D to 3D image conversion that incorporated image segmentation and depth estimation systems. They have created images from the left view and right view and displayed the stereo 3D image [3]. Murata et al. have developed a method for transforming 2D images of all kinds into 3D images. Adaptive use of the approach is to measure the depth of each separate region of the 2D images with their contrast, sharpness, and chrominance [4].

Cheng et al. have introduced an automatic system that converts 2D videos to 3D videos. Using the edge information, they grouped the regions into blocks and used bilateral filters to generate depth maps [5].

Zhang et al. have defined an approach that estimated a depth map by taking advantage of motion signals and photometric indications in video frames [6]. Su et al. have developed an algorithm for the real-time conversion from 2D to 3D. For 3D video formation, the 2D video accompanied by a depth image has been stored [7]. A hybrid algorithm for 2D to 3D conversion has been designed by Lai et al. For estimating depth, they have used motion knowledge, linear perspective, and texture characteristic. They have used bilateral filters for smoothing the depth map and eliminating noise [8].

The algorithms for the 2D-to-3D depth estimation have two problems. The first of these is the uniformity of depth within the object. Pixel grouping solves that problem. The second is a retrieval of a correct relationship of depth for all objects [9], [10]. When the object is moving with diverse vectors of self-motion, these strategies cause uncertainty concerning depth. Different depth values can be assigned to pixels belonging to the same entity. Thus, the method of estimating the depth map is an ill-positioned problem. In [11] and [12], 2D to 3D conversion method is developed based on edge information. In that method, the image is divided into blocks and depth maps are obtained.

To overcome the first issue, this paper presents a block-based image conversion method from which the edge information is calculated. Inspired by the block-based technique in video compression [13], this paper proposes a block-based technique in 2D to 3D conversion. This technique uses a grouping method in which the image is divided into blocks or groups depending on colors and spatial locality. Then the depth values are assigned to each group. A cross bilateral filter is then used to remove the blocking artifacts. After removing blocking artifacts, Depth Image Based Rendering (DIBR) is used to convert 2D images to 3D. Experimental results prove that the proposed algorithm works better than other recent methods. The computation time is very much reduced with block-based method.

The paper is organized as follows: Section II describes the overall system architecture of the proposed method. Section III elaborates all the phases in the proposed method such as block-based region splitting for calculating the edge, Depth map generation, and 3D image warping. Section IV demonstrates the proposed method with some experimental results followed by a conclusion in Section V.

II. Proposed System Architecture

This paper describes an efficient 2D-to-3D conversion method based on the use of edge information by block

S. Sowmyayani is with Department of Computer Science, St. Mary's College (Autonomous), Thoothukudi, Tamilnadu, India (e-mail: sowmyayani@gmail.com)

V. Murugan is with Department of Computer Science, MSU Constituent College of Arts and Science, Kadayanallur, Tamilnadu, India (e-mail: smv.murugan@gmail.com)

splitting. Most importantly, the edge of an image has more significance in generating a depth map. Once the pixels are split into blocks, a relative depth value can be assigned to each region. Next, the blocking artifacts created by the previous process are removed using cross bilateral filtering. Then, the multi-view images are obtained by the method of DIBR. Finally, the output 3D image is obtained without any blocking artifacts, thus enhancing the quality of the image in the display.

The overall system architecture of the proposed method is shown in Fig. 1. It consists of 4 important phases: Block Splitting, Depth Hypothesis, Filtering and DIBR.

The pixel values are converted to nodes and weights are calculated for all joining links. After this phase, a block is created from which Minimum Spanning Tree (MST) is constructed. Then the strong edges are removed to form multiple regions. For each region, depth values are assigned using the depth hypothesis. The blocking artifacts are removed using a cross bilateral filter. In the DIBR phase, the 2D image is converted to 3D images using depth values calculated in the previous phase. This phase consists of 3 sub-phases: Pre-processing, 3D Image Warping and Hole Filling. All the phases are elaborated in the subsequent section

III. Proposed Method Description

This section describes all the phases in the proposed methodology.

A. Block Splitting

The input image is split into blocks for calculating depth value. This implies that each pixel in the same block has the same depth value. In this paper, a 16x16 square-shaped block is used. The advantages of using block splitting are as follows:

1) It can retrieve appropriate depth value within an object.
2) It reduces computation time.

We consider an image of size 16 x 16. It is split into square shaped blocks by the following steps. The mean value is calculated for every 2 x 2 block and a node is created with the mean value. Thus, the block size is reduced to 8 x 8. After creating the node, the weights of the link are calculated by considering the absolute difference of the mean of neighboring blocks:

$$Diff(a,b) = \left| Mean(a) - Mean(b) \right| \quad (1)$$

where *a* and *b* are the two neighboring blocks. *Mean (a)* and *Mean (b)* represent the average color of block *a* and block *b* respectively. The smaller the value of *Diff(a,b),* the higher the similarity will be. Then the blocks are segmented into multiple regions using MST segmentation. The flow of the block-based region grouping method is shown in Fig. 2. Initially, an MST is constructed. Then multiple grouped regions or clusters are generated by removing the links of stronger edges in MST. In Figs. 2 (c)-(f), only the first 4 x 4 nodes are shown for clear understanding. Fig. 2 (f) shows the block with different depth values.

The number of links needed for the 16 x 16 square block is 480. After converting to node, the number of links is also reduced to 112. Thus, the computation time is slightly reduced. This information is given to generate a depth map.

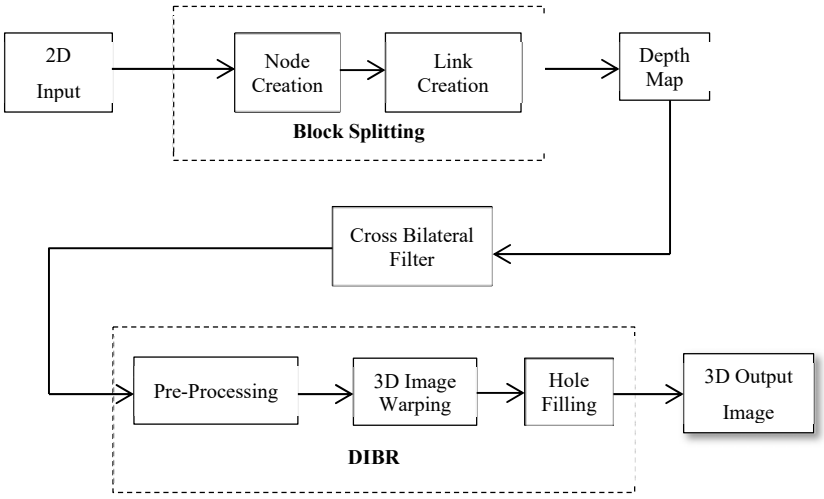


Fig. 1 Proposed System Architecture

B. Depth Hypothesis

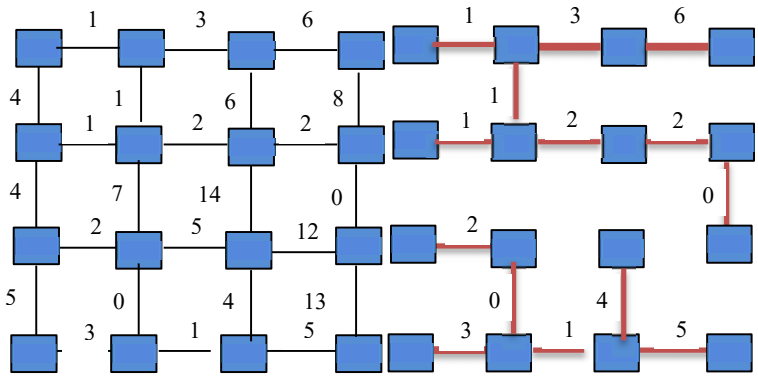
Depth extraction is the crucial one in the process of conversion. The difference between 2D and 3D images is the depth information, as mentioned earlier. Because of the depth information, the object will leap out of the screen and look like a real object. If the depth is extracted and incorporated, it will get 3D image. The algorithms of depth generation are loosely divided into three groups using different types of depth indications: binocular, monocular and pictorial depth indications. Each signal represents information varying in depth.

11	10	12	13	16	14	22	20	27	26	24	28	6	10	12	12
10	10	11	11	14	16	20	21	23	25	26	27	20	23	10	11
12	13	9	14	15	12	12	14	15	16	13	14	26	32	14	15
15	14	12	15	6	10	12	12	17	18	19	20	21	22	24	25
21	25	24	26	20	23	10	11	17	18	15	10	10	11	11	14
15	12	14	15	26	32	14	15	14	13	18	14	13	9	14	15
24	25	26	20	21	22	24	25	26	24	26	24	14	12	15	6
22	21	20	14	15	26	30	24	28	24	21	20	25	24	26	20
21	21	20	21	34	14	12	15	16	17	18	19	14	15	26	32
20	22	25	24	27	26	22	24	22	20	21	29	26	20	21	22
12	15	16	17	18	19	20	21	24	23	24	20	20	14	15	26
21	25	26	24	23	24	21	25	25	26	23	24	20	21	34	14
22	24	25	26	15	12	14	15	24	26	20	23	20	23	10	11
26	30	24	28	24	25	26	20	14	15	26	32	26	32	14	15
14	12	15	16	22	21	20	14	26	20	21	22	21	22	24	25
26	22	24	22	21	21	20	21	20	14	15	26	15	26	30	24

(a) Sample image pixels

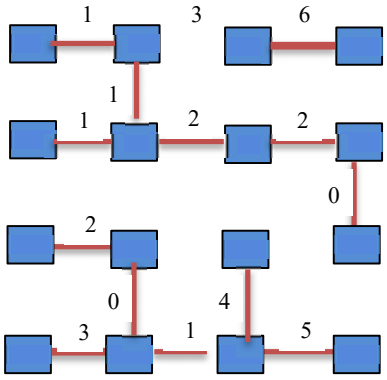
10	12	15	21	25	26	15	11
4	13	11	13	17	17	25	20
18	20	25	13	16	14	11	14
23	20	21	26	26	23	19	17
21	23	25	18	19	22	19	25
18	21	21	22	25	23	19	22
26	26	19	19	20	25	25	13
19	19	21	19	20	21	21	26

(b) Node Creation



(c) Link Creation

(d) MST



(e) Removing Strong Links

10	**12**	**15**	**21**
14	**13**	**11**	**13**
18	**20**	**25**	**13**
23	**20**	**21**	**26**

(f) Depth Values

Fig. 2 Flow of Block based Depth Estimation

In the 2D to 3D conversion process, the hypothesis of the depth gradient assigns the depth for each block. It involves the generation of gradient planes, the assignment of depth gradients, accuracy verification of the detected area, and finally the generation of depth maps. When each shift in the scene is detected, the linear scene perspective can be analyzed using Hough transform line detection algorithm [11]. The hypothesized depth gradient is given as:

$$Depth(R) = 128 + 255\left\{\sum_{pixel(x,y)} W_{rl}\frac{x-\frac{width}{2}}{width} + W_{ud}\frac{y-\frac{height}{2}}{height}\right\} / \ pixel\_num(R) \quad (2)$$

where $|W_{rl}| + |W_{ud}| = 1$.

$$Depth_{(x_i)} = \frac{1}{N(x_i)}\sum_{x_j\in\Omega(x_i)} e^{-0.5\left[\frac{|x_j-x_i|}{\sigma_x^2}+\frac{|u(x_j)-u(x_i)|^2}{\sigma_y^2}\right]} Depth(x_j) \quad (3)$$

$$N(x_i) = \sum_{x_j\in\Omega(x_i)} e^{-0.5\left[\sigma_x^2|x_j-x_i|+\sigma_y^2|u(x_j)-u(x_i)|^2\right]} \quad (4)$$

A greater depth value implies that the pixel is closer to the user. Equation (2) implies that that the depth value is the center of gravity of the block group i.e. each pixel in the group belongs to the same depth value. The $|W_{rl}|$ and $|W_{ud}|$ can be adjusted to the left-to-right and top-to-bottom depth gradient. The direction of the concept of the depth gradient can be extracted from the study of the images from a geometrical perspective. Results of an analysis [14] indicate that the bottom-up mode is the most important mode in the real world. If the scene mode is not detected by the linear perspective information, then the bottom-up mode is the default mode to choose.

C. Cross Bilateral Filtering

The bilateral filter is non-iterative and achieves acceptable results with only one run. It makes the parameters of the filters fairly straightforward as they do not cumulate their effects over more iteration. Though it is slow, the bilateral filter is proven to be very useful. It is nonlinear and its evaluation is also computationally expensive. Traditional methods, such as performing convolution after a Fast Fourier Transform, are not applicable. Nonetheless, approaches were suggested later to improve the bilateral filter assessment. Unfortunately, these methods appear to be based on approximations that are not based on mathematical foundations.

The proposed method has selected the cross bilateral filtering from among the variants of the bilateral filter. In certain applications, such as computational photography, it is also useful to decouple the data that need to be smoothed to identify the edges that need preservation. A version of the classical bilateral filter is the cross bilateral filter. This filter is used to smooth out the image to find the preservation edges. The depth map created by grouping of block-based regions contains blocky objects. Here the blocky objects are removed using the cross bilateral filter.

Here the cross bilateral filter finely smoothed the depth map while maintaining the boundaries of the objects [14], [15]. The blocky artifact is effectively removed in the created depth map while the sharp discontinuities of depth along the boundary of the object are retained.

D. Depth Image-Based Rendering

The filtered depth map has good visual quality since the cross bilateral filter produces a smooth depth map with identical pixel values within the smooth region and retains sharp discontinuity on the boundary of the artifacts. Using DIBR for 3D visualization [16], the depth map is then used after filtering by the cross bilateral filter to produce left/right or multi-view images. It includes three sub-phases: pre-processing of the depth map, 3D image Warping and Hole-Filling. A Smoothing filter is the first stage to smooth the depth map. Then, according to the smoothed depth map and also intermediate view, the 3d image warping produces left and right views. If the picture still contains holes, then hole-filling is added to fill these holes with color.

E. Pre-Processing of Depth Image

Depth image preprocessing is typically a smoothing filter. Since depth image with that of the sharp horizontal transition can result in large holes after warping, smoothing filter is applied to smooth sharp transition to reduce the number of large hole. However, if depth image is blurred, not only large holes are reduced but also the blurred view is degraded as the depth map of the non-hole region is smoothed out.

F. 3D Image Warping

This process maps the pixel of the intermediate view to left or right view according to the pixel depth value. In other words, the 3D image warping mechanism transforms the pixel position according to the depth value. The 3D image warping formula is:

$$\begin{aligned} x_l &= x_c + \left(\frac{t_x}{2}\frac{f}{Z}\right) \\ x_r &= x_c - \left(\frac{t_x}{2}\frac{f}{Z}\right) \end{aligned} \quad (5)$$

where, x_l, x_r and x_c are the horizontal coordinates of the left, right and intermediate view. Z is the depth value of current pixel, f is the camera focal length and t_x is the eye distance. It implies that, in horizontal direction, 3D warping maps pixels of the intermediate view to one of the left and right view.

G. Hole Filling

Average interpolation filter method is a popular method for DIBR Hole-Filling. The average filter will, however, result in highly textured areas having artifacts. In addition, hole size in DIBR is so huge that average filter with a large window size is required. At the same time, edge information cannot be preserved by the typical filter with a wide window size. Hence, the edge information is blurred.

IV. Experimental Results

The experimental results of the proposed method are compared with methods of [4] and [5]. The proposed method is tested on an image of size 450 x 375. The input image, depth map and the output image are shown in Figs. 3 and 4 respectively.

For our quantitative evaluation, three commonly-used metrics such as average relative error (rel), average log10 error (log10), root mean squared error (rms), Peak Signal to Noise Ratio (PSNR), accuracy and computation time are used.

$$rel = \frac{1}{T}\sum_p \frac{\left|d_p^{gt} - d_p\right|}{d_p^{gt}} \tag{6}$$

$$\log 10 = \frac{1}{T}\sum_p \left|log_{10} d_p^{gt} - log_{10} d_p\right| \tag{7}$$

$$rms = \sqrt{\frac{1}{T}\sum_p (d_p^{gt} - d_p)^2} \tag{8}$$

The PSNR is most commonly used as a measure of quality of image reconstruction. It is defined as

$$PSNR = 20log_{10}\left(\frac{MAX_I}{\sqrt{MSE}}\right) \tag{9}$$

Here, MAX_I is the maximum possible pixel value of the image.

Fig. 3 Input 2D Image

The MSE is given as

$$MSE = \frac{1}{T}\sum_p \left|(d_p^{gt} - d_p)\right| \tag{10}$$

The accuracy under a threshold [7]

$$\max\left(\frac{d_p^{gt}}{d_p}, \frac{d_p}{d_p^{gt}}\right) = \delta < th \tag{11}$$

(a)

(b)

Fig. 4 (a) Depth Map (b) Output 3D Image obtained by the proposed method

where d_p^{gt} and d_p are the ground-truth and predicted depths at pixel indexed by p. T is the total number of pixels in all the evaluated images and th is a predefined threshold. Table I compares the results obtained by the proposed method with other methods. The average runtime of the proposed method is very less than 5 seconds. Table II shows the PSNR and accuracy obtained by the proposed method and other methods.

From Table I, it is clear that the proposed method achieves very less error rate when compared to other methods. An increase in PSNR and accuracy is achieved by the proposed method. The performance of the proposed method is also compared with state-of-the-art single image depth estimation methods for the NYU-V2 dataset [17]. This dataset comes with hand-labeled semantic segmentation annotations.

The results of [18]-[20] are provided for comparison. The system of [19] predicts the same resolution as the ground truth, while the other two methods make depth prediction at lower resolution. Table III reports the comparison of all methods for NYU-V2 dataset. From Table III, it is observed that the proposed method has lesser error and higher accuracy than other recent methods in NYU-V2 dataset.

TABLE I
PERFORMANCE COMPARISON OF THE PROPOSED METHOD WITH RECENT METHODS IN MAKE3D DATASET

Methodology/ Metrics	Rel	LOG 10	RMS
Murata et al. [4]	0.336	0.124	9.55
Cheng et al. [5]	0.355	0.132	9.42
Proposed Method	0.331	0.117	9.21

TABLE II
ACCURACY AND PSNR COMPARISON OF THE PROPOSED METHODS WITH OTHER METHODS IN MAKE3D DATASET

Methodology/ Metrics	PSNR (dB)	Accuracy $\delta <$		
		1.25 (%)	1.25^2(%)	1.25^3 (%)
Murata et al. [4]	19.25	99.95	99.98	99.99
Cheng et al [5]	20.01	99.95	99.97	99.99
Proposed Method	20.54	99.95	99.98	100.0

It is observed that from Table II, the higher the threshold value, the higher the accuracy. When $\delta < 1.25^3$, the accuracy reaches its maximum value.

TABLE III
COMPARATIVE RESULTS ON NYU-V2 DATASET

Method/ Measure	Rel	RMSE		Accuracy $\delta <$		
		Linear	Log 10	1.25 (%)	1.25^2 (%)	1.25^3 (%)
Eigen et al. [18]	0.144	0.75	0.210	62.6	89.9	97.6
Liu et al. [19]	0.143	0.64	0.206	67.6	92.1	98.1
Eigen and Fergus [20]	0.139	0.63	0.192	70.9	91.9	98.0
Proposed Method	0.121	0.59	0.186	72.3	92.4	98.4

From Table III, it is observed that the proposed method has lesser error and higher accuracy than other recent methods in NYU-V2 dataset. When threshold $\delta < 1.25$, the accuracy achieved by the proposed method is only 72.3%. But when it is increased to 1.56 (i.e. 1.25^2) and 1.95 (i.e. 1.25^3), the accuracy is also increased to 92.4% and 98.4% respectively.

V. CONCLUSION

In this paper, a method is proposed for converting 2D to 3D images. It uses block-based technique to reduce computation time. The proposed method is compared with two recent methods which use grayscale based 2D to 3D conversion method and the second one uses square size blocks for depth estimation. From the experimental evaluation, it is evident that the proposed method works better than other recent methods in terms of quantitative metrics. Also, the computation time is very much reduced by the proposed method. The proposed method achieves the highest accuracy of 100% which is better than other depth estimation methods.

REFERENCES

[1] W. J. Tam, and L. Zhang, "3D-TV content generation: 2D-to-3D conversion," in Proc. ICME, pp. 1869-1872, 2006.
[2] X. Y. Mao and L. K. lbsiyasu, "Hierarchical representations of 2D/3D Gray-Scale Images and their 2D/3D two way conversion," IEEE, pp. 37-44, 1987.
[3] T. L. Chin, C. L. Chin, K. W. Fan, and C.Y. Lin, "A novel architecture for converting single 2D image into 3D effect image," IEEE, pp. 52-55.
[4] H. Murata, X Mori, S. Yamashita, A. Maenaka, S. Okada, K. Oyamada, and S. Kishimoto, "A real-time 2-D to 3-D image conversion technique using computed image depth," SID Symposium Digest of Technical Papers, vol. 29, no. 1, pp. 919-923, 1998.
[5] C. C. Cheng, C. T. Li, and L. G. Chen, "A 2D-to-3D conversion system using edge information," in Proc. Digest of Technical Papers International Conference on Consumer Electronics, 2010, pp. 377-378.
[6] Z. B. Zhang, Y. Z. Wang, T. T. Jiang, and G. Wen, "Visual pertinent 2D-TO-3D video conversion by multi-cue fusion," in Proc. 18th IEEE International Conference on Image Processing, 2011, pp. 909-912.
[7] C. L. Su, K. N. Pang, T. M. Chen, G. S. Wu, et al., "A real-time Full-HD 2D-to-3D conversion system using multicore technology," in Proc. fifth FTRA International Conference on Multimedia and Ubiquitous Engineering, IEEE, 2011, pp. 273-276.
[8] Y. K. Lai, Y. F. Lai, and Y. C. Chen, "An effective hybrid depth-generation algorithm for 2D-to-3D conversion in 3D displays," Journal of Display Technology, vol. 9 no. 3, pp. 154-161, March 2013.
[9] Y-L. Chang, et al, "Depth Map Generation For 2D-To-3D Conversion by Short-Term Motion Assisted Color Segmentation" in Proceedings of ICME, 2007.
[10] D. Kim, D. Min, and K. Sohn, "A Stereoscopic Video Generation Method Using Stereoscopic Display Characterization and Motion Analysis", in IEEE Trans. On Broadcasting, Vol. 54, Issue 2, pp. 188-197, 2008.
[11] C. C. Cheng, C. T. Li, P. S. Huang, T. K. Lin, Y. M. Tsai, and L. G. Chen, "A block-based 2D-to-3D conversion system with bilateral filter" in Proceedings of IEEE International Conference on Consumer Electronics, 2009.
[12] S. Bharathi, A. Vasuki, "2D-To-3D Conversion of Images using Edge Information" International Journal of Computer Applications.
[13] S. Sowmyayani, P. Arockia Jansi Rani, "Block based Motion Estimation using Octagon and Square Pattern", International Journal of Signal Processing, Image Processing and Pattern Recognition, 2014, Vol. 7, Iss. 4, pp.317-324.
[14] S. Paris and F. Durand, "A fast approximation of the bilateral filter using a signal processing approach", in MIT Technical Report (MIT-CSAIL-TR-2006-073), 2006
[15] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in Proc. ICCV, pp. 839-846, January 1998.
[16] W.-Y. Chen and Y.-L. Chang and S.-F. Lin and L.-F. Ding and L.-G. Chen." Efficient depth image based rendering with edge dependent depth filter and interpolation," in Proc. ICME, pp. 1314-1317, 2005.
[17] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus. Indoor Segmentation and Support Inference from RGBD Images. In ECCV, 2012.
[18] D. Eigen, C. Puhrsch, and R. Fergus. Depth map prediction from a single image using a multi-scale deep network. In Advances in Neural Information Processing Systems 27, 2014.
[19] F. Liu, C. Shen, and G. Lin. Deep convolutional neural fields for depth estimation from a single image. In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.
[20] D. Eigen and R. Fergus. Predicting depth, surface normal and semantic labels with a common multi-scale convolutional architecture. In Int. Conference on Computer Vision, 2015.