International Journal of Engineering & Technology, 7 (2.6) (2018) 28-32

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research Paper

Non parametric methods of disparity computation

Priya Charles [1] *, A. V. Patil [2]

[1] *Electronics &Telecommunication department, VIIT, Pune, India-411044*
[2] *DYPIEMR, Pune, India-411044*
**Corresponding author E-mail: prinnu@yahoo.com*

Abstract

Disparity is inversely proportional to depth. Information about depth is a key factor in many real time applications like computer vision applications, medical diagnosis, model precision etc. Disparity is measured first in order to calculate the depth that suits the real world applications. There are two approaches viz., active and passive methods. Due to its cost effectiveness, passive approach is the most popular approach. In spite of this, the measures are limited by its occlusion, more number of objects and texture areas. So, effective and efficient stereo depth estimation algorithms have taken the toll on the researchers. The important goal of stereo vision algorithms is the disparity map calculation between two images clicked the same time. These pictures are taken using two cameras. We have implemented the non-parametric algorithms for stereo vision viz., Rank and Census transform in both single processor and multicore processors are implemented and the results showsits time efficient by 1500 times.

Keywords: *Stereo Image; Disparity; Depth; Non Parametric.*

1. Introduction

One most important sense organ for humans is the Vision. We see the 3D world with our naked eyes and that helps us to move on in our day to day lifestyle and work. But when we click images of a 3Dworld system this third dimension of depth gets lost hence we are unable to compute the actual distance of the objects from the point where sensors were placed and also the relative distance between objects. Stereo vision is a famous and proper approach among the advanced machine vision bureaucrats. With advancement in the computer vision and high speed computing units we are now able to compute this third dimension which was lost earlier and this method of computing the lost depth is known as Depth Estimation.

It is even called a reliable tool to extract the depth from a particular scene. How accurate they are purely depends on the quality of the cameras and the correspondence algorithm used for extracting the disparity pixels A stereo matching algorithm matches pixel values of input image with the query image and explores the corresponding vertical and horizontal displacement as the disparity value, which is inversely proportional to its depth and hence it can retrieve the 3rd dimension viz., the depth.

As we see movies in 3D are increasing in faster rate, even televisions give the TV audience the opportunity to experience 3D, we see tremendous scope of improvement in stereovision area for depth calculation . Basically, the 3D experience comes from the left and right eye seeing slightly different views of the same scene. 3D view is perceived by disseminating two separate views for the left and right eye. Stereo vision is taking the click of a scene with two cameras positioned at some distance. This distance is usually 50mm the minimum which is the distance between the two eyes of a human being, in order to replicate the effect in robots. Figure 1 shows the sample of original, the left image and the right images. This shows the effect of pictures taken from 2 cameras at some distance.

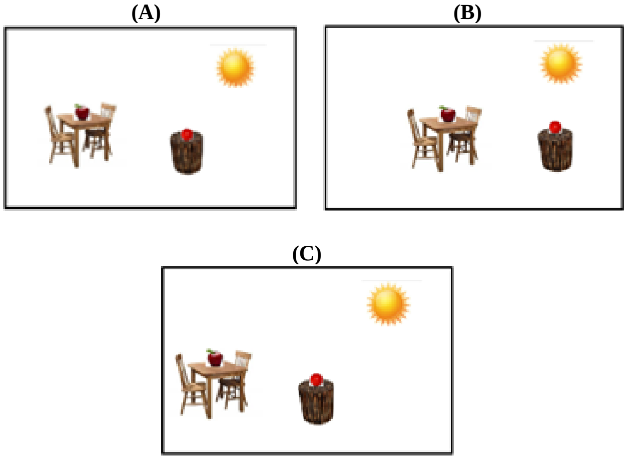

Fig. 1: A) Image at Actual. B) Input Image. C) Query Image.

Objects closer to the camera is towards the right in the input image, and towards the left in the query image. Distant objects will be located at approximately the same place location in both images. By comparing these two images, the displacement of objects between the two images gives disparity information. The depth is calculated by taking the inverse of disparity. Hence comes the goal of finding the matching pixel between the images.

Stereo correspondence algorithms are classified based on the output which could be a dense or sparse output. The methods that use segments and edges produces sparse outputs but they have a good combination of speed and accuracy.

For real time applications/robotics demand dense output. We have aimed at algorithms for real time applications and hence we have focused on dense stereo correspondence algorithms. There are two types of local methods viz., area based and feature based methods. Local methods (area-based) [5] are also called as window based methods because the disparity in this case for a given point depends purely on the intensity values of that particular window. The window size can be changed depending on the accuracy of the output. This method gives a dense output that suits the real world applications. The feature based methods rely on the feature extraction, and gives a sparse output but the speed relatively is high. Global methods are otherwise called energy based methods are very accurate but time consuming and computationally expensive.

The algorithms here would be implemented with the assumption that the same object/feature in both input and query images have the same intensity values. This assumptions may sometimes go invalid for the reason that due to different angles of cameras the intensities for the same pixel point may be different because of illumination effects. Stereo vision algorithms normally fail in non-ideal lightning conditions in spite of the two cameras taken into consideration are perfectly tuned. It because of the reason that though the images are taken at the same instant the pose or orientation of the camera might be in such a manner that intensity of light varies for the two images captured.

The whole concept is about matching the relevant pixel. In order to match the pixels in the pair , the pixels are to be searched in both the input and the query images. Since the pair of images are taken by keeping some distance in between them there are chances that the variation could be possibly in x and as well as in y direction. If this is the case then the search becomes difficult.

In order to avoid this we go for image rectification. The idea of image rectification is to make epipolar lines of two camera images horizontally aligned. This is done using linear transformations like rotate, translate and skew of the camera images. This aligns the images in one direction.

We propose the use of image rectification after the calibration of the camera. Stereo vision algorithms normally fail in non-ideal lightning conditions in spite of the two cameras taken into consideration are perfectly tuned.

Section 2 provides process block diagram with its explanation. Section 3 presents the performance difference of CPU and GPU, Section 4elaborates the conclusion.

2. Process block diagram

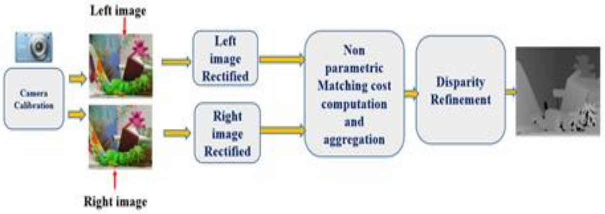

Fig. 2: Block Diagram.

2.1. Camera setup and calibration

Stereo images have to be clicked of the same scene using two cameras from two viewpoints. Here we have used a single camera and a slider on which the camera is mounted to click images as shown in figure 1. Camera is placed on a tripod stand and camera is mounted on top of it using a slider. A left image is clicked and then slider is moved to click right images by changing the distance in mm with a start of 50 mm and ended with 200 mm [3].

Multiple image datasets can be created using this setup in varying surroundings such as indoor, outdoor, near and far. These created datasets can then be used for computing depth of the real world scene.

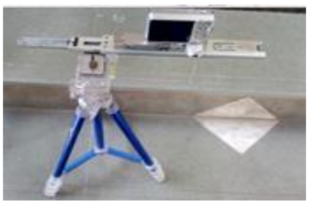

Fig. 3: Camera Setup for Capturing Images.

Camera calibration is another important step which helps in removing errors from the captured stereo images and improves accuracy of the result. When we capture images from two different viewpoints or using two different cameras shifted by some distance as mentioned in previous paragraph then different errors come into picture like rotational error, tangential error etc. In order to remove these unknown errors there is a need to compute Intrinsic and Extrinsic parameters of the stereo camera system and use these parameters to calibrate the camera system.

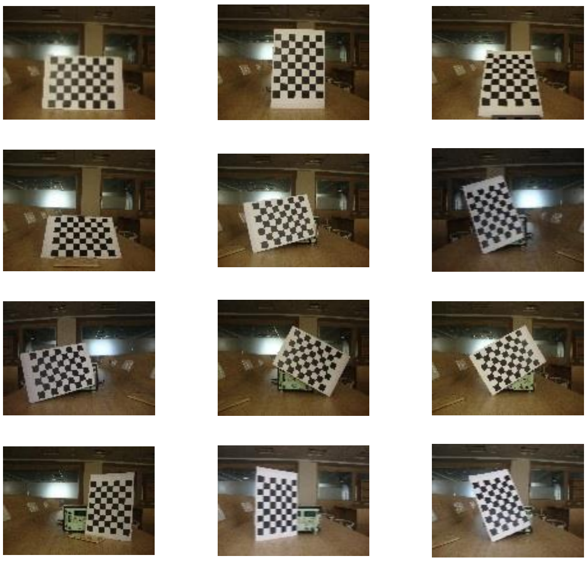

Fig. 4: A) Left 12 Images of Different Angles and Positions.

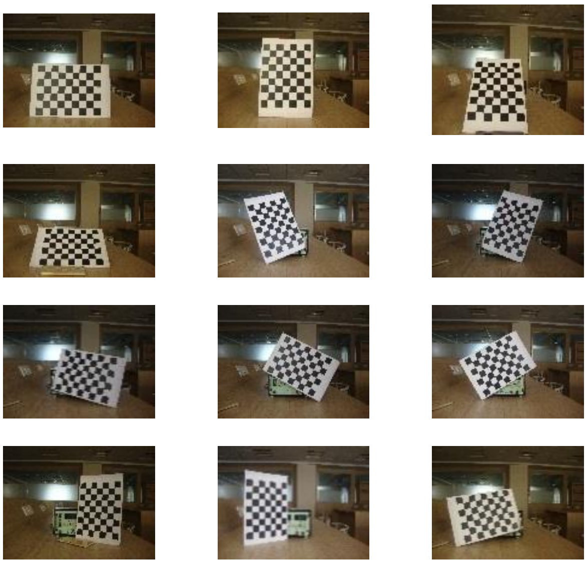

Fig. 4: B) Right12 Images of Different Angles and Positions.

Here we have used a checkerboard of dimension 31.5cm X 21cm with each block of sixe 2.8cm X 2.8cm as shown in figure 2a and 2b. 24 images are captured with this setup which includes 12 left stereo images and 12 right stereo images.

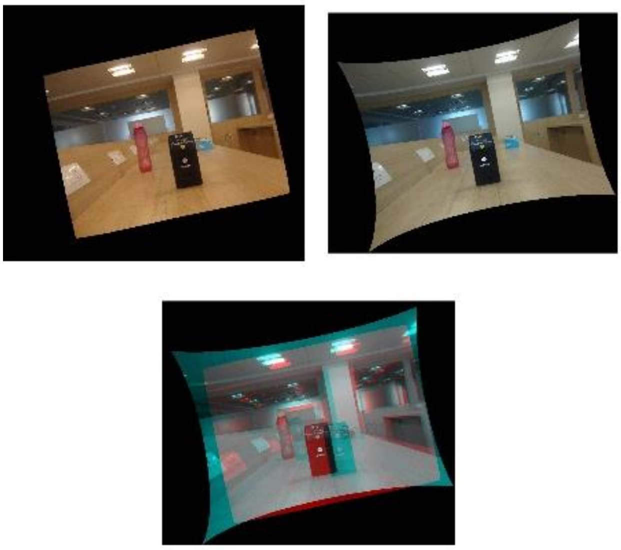

Fig. 5: Calibrated and Rectified Input and Query Images and the Combined Images Respectively.

These images were given to the Matlab2016 stereo camera calibrator app, for which Intrinsic and Extrinsic parameters were obtained. These parameters are then used for calibration and rectification. Hence the stereo matching problem is reduced to 1 dimension.

2.2. Non parametric local transforms

Non parametric local transforms depend on the order of arrangement of intensity values and not on the intensity values itself [6]. This enhances the performance of the near object boundaries. In this paper we have used two nonparametric local transforms viz.,the rank transform and the census transform. The former measures the local intensity and the latter summarizes the logic image structure. Most of the approaches to the correspondence problems have problems near the boundaries due to discontinuities in disparity. This was the major problem with parametric transforms.
The rank transform algorithm is as follows:

1) Read the left image.
2) Iterate for every pixel around a particular fixed window, choose a reference pixel and check whether the neighboring pixels are less the reference pixel .All the neighboring pixel values are given the rank from 1 onwards and saved in the rank matrix.
3) Save this left rank matrix and do the same for right image
4) Now we get two rank matrices, compute the distance by iterating from 1 to 8 (8 being number of bits).
5) Find the sum of the distance.
6) Find the minimum sum value and store its respective disparity range in a disparity matrix.

The census transform algorithm is as follows:

1) Take the left image and rotate a window around a fixed pixel.
2) For every pixel in that window with reference to the centre pixel compute new value using formula if pixel intensity <ref pixel then the value would be 1 else 0. Do the same for right image.
3) Now compute minimum hamming distance using XOR and fin the total sum of distance for each disparity value.
4) The disparity value at which we get minimum hamming distance, is stored in the disparity matrix.

The algorithms that are implemented in this paper considers an known disparity search range, by manually finding the disparity range, since it is crucial for stereo matching tasks, to find the known disparity range using algorithms as it is a time consuming task. It also does not perform well. In the case of real time image processing two aspects have to be observed while designing: time efficiency and results accuracy.

The disparity range defines the minimal and maximal disparity in the given stereo image pair by manually searching the row and column of the stereo pairs to find the maximum disparity which might not be very rigorous. Many algorithms decrease their performance for an unknown disparity search range. Well-known and benchmark stereo image pairs are typically provided with already given disparity search range [2]. However, we usually want our algorithms to be run on our data sets and images where we have no information available.

3. CPU VS GPU

CPU coding of both Rank and Census transforms are done using MATLAB. Single core CPUs are meant to have single thread at one time. So their time complexity is high.
CPU multi core is not used as the maximum of around 12 cores (Intel), and as the application is related to images which are of huge size we have implemented them with GPU cores [15].
In order to compare it with multi core we have used GPUs with many threads which can parallel run many threads. A GPU has hundreds or thousands of cores running at once. In the sense same code runs on all the threads the same time. ration only one at a time which is a "Kernel" on the entire image array. The scheduler launches thousands of threads one for each core and then executes the kernel [15].
CUDA capable GPU: CUDA is NVIDIA's parallel computing architecture that enables huge increases in computing performance by harnessing power of the GPU (graphics processing unit). CUDA capable GPU's allow to do Parallel programming, it allows to launch multiple parallel Threads which speed up the computation.
To run a CUDA program one must need a machine running a CUDA capable GPU. The GPU model that is used is QUADRO K1100M with CUDA toolkit 7.5.
The number of threads we have used is based on the size of the image and is standardized using the formula

```
column = (blockIdx.x * blockDim.x) + threadIdx.x;

row = (blockIdx.y * blockDim.y) + threadIdx.y;
```

The dimension of the threads and blocks are calculated in the following way

```
dim3 threads (32, 32);

dim3 blocks (w/threads.x, h/threads.y);
```

where w and h are the width and height of the image. We have chosen to use 32 X32 thread in each block. Hence No of blocks used is 14X11.14 in the horizontal blocks and 11 as the vertical blocks.
The Warp size for this GPU model is 32, Maximum number of threads possible is 2048 and maximum number of threads per block is 1024. [15]

3.1. Evaluation methodology

Quality measures are one most important standard to measure the accuracy of the algorithm by comparing the results with existing ground truths [9]. To evaluate the performance of a stereo algorithm or the outcome with the change in parameters, a quantitative way is to be used to estimate the performance of the computed correspondences [13].
The two methods considered in this paper are RMS error (Root Mean Square error) and BAD PIXEL Match.

1) RMS (root-mean-squared) error: (measured in disparity units) between the computed disparity map dC(x,y) and the ground truthmapdT (x, y) [13], i.e.

$$R= sqrt\ [\ 1/N\sum(x,y)\ [dC(x.y)-dT(x,y)\ ]\ ] \tag{1}$$

Where N is the total number of pixels.

2) Percentage of bad matching pixels: This quality metric gives percentage of mismatching pixels in the computed disparity map and the ground truth.

$$B = [\ 1/N\sum(x,y)\ [dC(x.y)-dT(x,y)\ ] > \lambda d\ ] \qquad (2)$$

Where λd (eval bad thresh) is a disparity error tolerance.
For the experiments in this paper we use λd =1 .0, since this coincides with some previously published studies [4] and [5].

3.2. Simulation results

The input images are taken from online Middlebury Stereo Dataset along with their ground truths [4]. Disparity maps for Rank and Census transforms are computed for those Middlebury datasets [4] and our results both rectified and non-rectified are compared with their ground truths available online to see how accurate are the results obtained. Our own data base results are also displayed.
The actual distance was measured and the Digital Camera with Specifications-Brand Sony, Product Line Sony Cyber-shot, Model DSC-W220, Sensor Resolution 12.1 Megapixel Optical Sensor Size 1/2.3" was used to generate data base with 50 mm ,75mm,100mm,150mm cameras apart.
Figures and Tables are shown below which show quantitative comparison of these methods among each other and online available datasets.

3.3. Rank transform results: CPU

a) Cones database (2003)

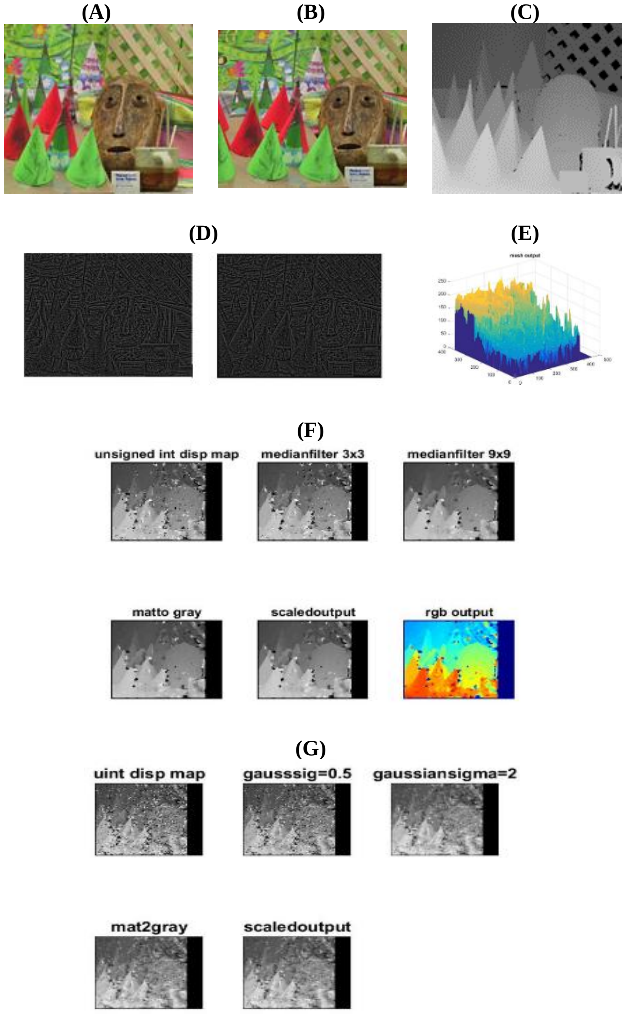

Fig. 6: (A)CONES Dataset. (B) Input Left and Right Image. (C) Ground Truth. (D) Rank Left and Right. (E) Mesh. (F) Refinement Using Median Filter. (G) Refinement Using Gaussian Filter.

b) Teddy database (2003)

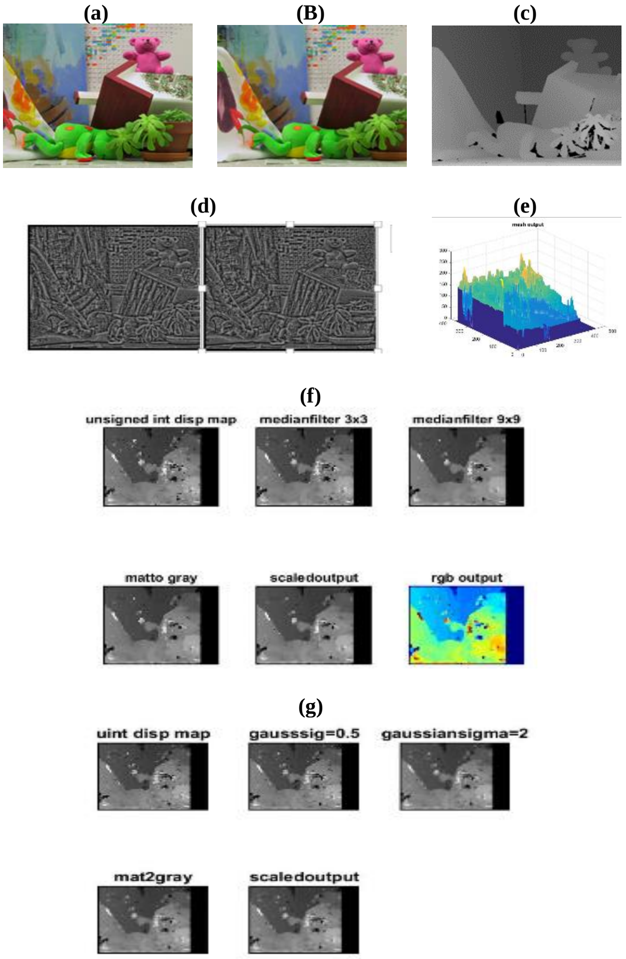

Fig. 7: TEDDY Dataset. (A) And (B) Input Left and Right Images. (C) Ground Truth. (D) Rank Left and Right. (E) Mesh. (F) Refinement Using Median Filter. (G) Refinement Using Gaussian Filter.

c) Census transform results: CPU

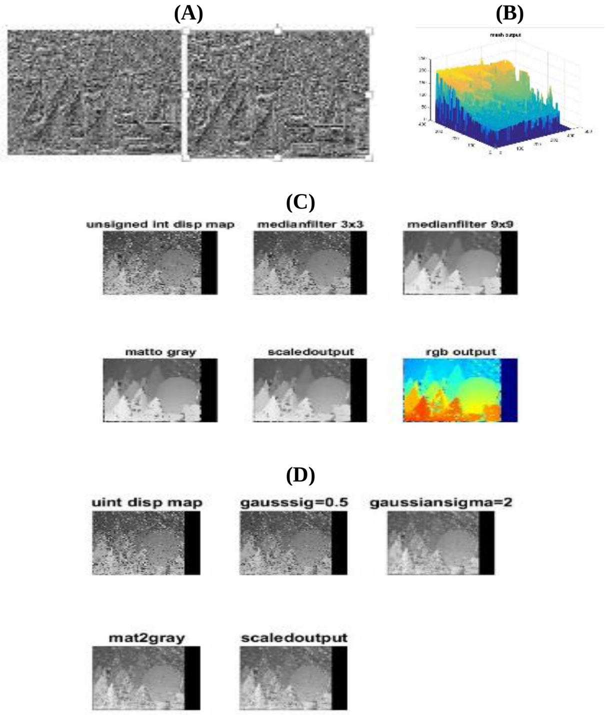

Fig. 8: (A) Census Left and Right. (B) Mesh. (C)Refinement Using Median Filter. (D) Refinement Using Gaussian Filter.

d) rank and census transform results: GPU

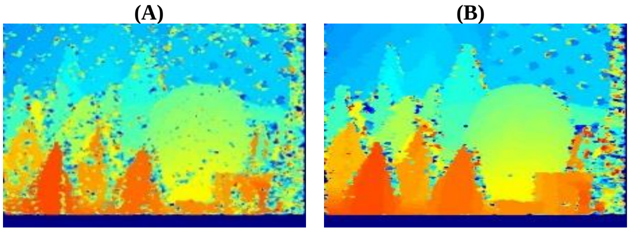

Fig. 9: (A) Rank. (B) Census.

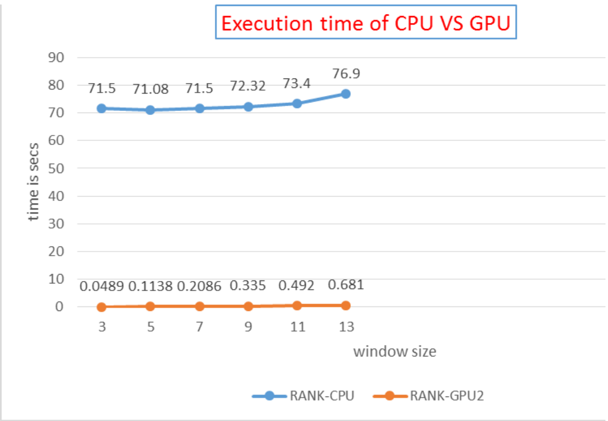

Fig. 10: Execution Time for Image Size 375X450.

4. Conclusion

Non parametric methods work approximately same for normal scenes but work very fine for texture-less surfaces where parametric algorithm fail. The time taken for CPU is 1500 times more than that of a GPU for the same image. This work can be extended for texture based images with occlusion in future keeping the performance and time complexity as the focus.

Acknowledgement

The authors would like to thank and show gratitude to Dr. D. Y. Patil Institute of Engineering, Research and Management, Akurdi for sharing their research facilities during the course of this research. I would not forget to remember my colleagues who helped me to materialize the research.

References

[1] Scharstein, D., &Szeliski, R. (2002), "A taxonomy and evalua-tion of dense two-frame stereo correspondence algorithms" In-ternational Journal of Computer Vision, 47(1-3), 7–42. https://doi.org/10.1023/A:1014573219977.

[2] Jana Kostkov´a and Radim ˇS´ara," Automatic Disparity Search Range Estimation for Stereo Pairs of Unknown Scenes",Center for Machine Perception, FEE, Czech Technical University Karlovon´am. 13, 121 35 Prague, Czech Republic

[3] Schuon, S., Theobalt, Ch., Davis, J., &Thrun, S. (2008)." High-quality scanning using time-offlight depth super resolution" In: IEEE CVPR Workshop on Time-Of-Flight Computer Vision 2008, pp. 1-7

[4] R. Szeliski and R. Zabih, "An experimental comparison of ste-reo algorithms. In International Workshop on Vision AlgorithmsKerkyra, Greece, 1999. Springer,", pages 1–19

[5] C. L. Zitnick and T. Kanade." A cooperative algorithm for stereo matching and occlusion detection". IEEE TPAMI, 22(7) 2000, pgs 675–684,

[6] S. Birchfield and C. Tomasi.," Depth discontinuities by pixelto-pixel stereo". In ICCV, 1998, pages 1073–1080,

[7] Ozden, K. E., Schindler, K., and van Gool, L. (2007)." Simulta-neous Segmentation and 3D Reconstruction of Monocular Im-age Sequences. Computer Vision", 2007. ICCV 2007. IEEE 11th International Conference on, pp. 1-8.

[8] Helmi, F. S. & Scherer, S. (2001)" Adaptive Shape from Focus with an Error Estimation in Light Microscopy", 2nd Int'l Sym-posium on Image and Signal Processing and Analysis, pp. 188-193.

[9] Nalpantidis, L., Chrysostomou, D., &Gasteratos, A. (2009, De-cember). "Obtaining reliable depth maps for robotic applica-tions with a quad-camera system. In International Conference on Intelligent Robotics and Applications" (vol. 5928, p. 906-916). Singapore: Springer-Verlag https://doi.org/10.1007/978-3-642-10817-4_89.

[10] T. Kanade." Development of a video-rate stereo matching". In Image Understanding Workshop, Monterey, CA, 1994. Mor-gan Kaufmann Publishers, pages549–557.

[11] M. J. Hannah. "Computer Matching of Areas in Stereo Images". PhD thesis, Stanford University, 1974

[12] T. W. Ryan, R. T. Gray, and B. R. Hunt. "Prediction of correla-tion errors in stereo-pair images". Optical Engineering, 1980,19(3):312–322,

[13] Viral H. Borisagar, Mukesh A. Zaveri, Disparity Map Genera-tion from Illumination Variant Stereo Images Using Efficient Hierarchical Dynamic Programming",The Scientific World Journal, Vol-ume 2014 (2014), Article ID 513417, 12 pages

[14] Haesol Park, Kyoung Mu Lee, "Joint Estimation of camera pose, Depth, Deblurring and super Resolution from a blurred image sequence", 2017 IEEE Internation conference on Computer vi-sion 2380-7504/17$31.00© IEEE pgs 4623-4631.

[15] Adrian Leu, Dan Bacără, IoanJiveţ," Disparity Map Computa-tion Speed Comparisonfor CPU, GPU and FPGA Implementa-tions", Tom 55(69), Fascicola 2, 2010, pgs 7-12.