An Efficient Algorithm for Depth Map Upscaling

Somashailaja. M[#1], Dr. M. Z. Kurian [*2]
Sri Siddhartha Institute of Technology, Tumkur, Karnataka, India – 572104
[1] M.Tech, Digital Electronics
[2] Professor and H.O.D, Department of E&C

Abstract**—** Depth information is used in various image processing applications. Depth information can be recovered from depth map captured from depth sensor. Captured depth map will be noisy and low resolution. Hence this work is devoted for upscaling the depth map for further future application using 'Patch Extraction' technique. MATLAB R2013a is used for programming.

Keywords**— Depth Map, Depth Sensors, Depth Upscaling, Bicubic interpolation, Patch Extraction Algorithm, MATLAB R2013a.**

I. INTRODUCTION

Depth information of an object or scenery is used in image processing applications in gaming and entertainment industry, security and surveillance, space application, automatic robot navigation and industry automation. The required depth information can be captured from depth sensor which are specially designed for depth calculations.

Available depth sensors like Swiss Ranger, PMD CamCube, Microsoft Kinect to name a few, in the market are of low resolution. Beside this, captured depth map sometimes affected by noise. For the above mentioned applications depth map should be noise free and high resolution. For example in automatic robot navigation low resolution does not matters but depth map should be noise free. Presence of noise creates fault depth readings which in turn misleads robot while navigating which ends up in collision. In automation industry, application requires zooming of depth map which in turn demands high resolution and noise free depth map. In entertainment industry like 3D cinemas and 3D TV, depth map resolution should match the corresponding high resolution color image. Here high resolution depth map has crucial role. Hence this work tries to denoise and upscale the captured depth map. Fig.1 is a depth map and its corresponding color image.

(a) (b)

Fig.1: (a) Depth Map (b) Corresponding Color image.

II. LITERATURE SURVEY

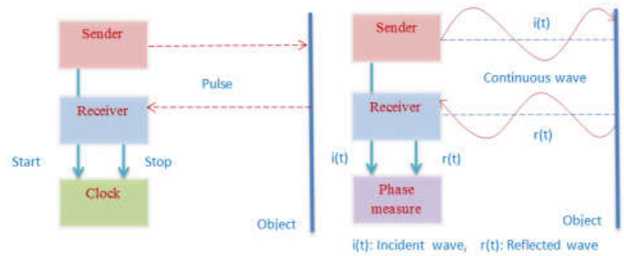
Before getting into the actual topic of depth map upscaling let us discuss the preliminary step i.e depth map capturing. There are two ways by which depth map can be captured. One is Stereo analysis depth capturing and the other one is Sensor based depth capturing

Stereo analysis is an old technique of depth calculation where a scene is captured from two or more color cameras as shown in fig.2. Depth map is estimated based on some of the camera parameters like camera position, orientation, resolution, focal length, pixel aspect ratio etc. This technique gives high resolution depth map but the downsides are, tedious multiple camera alignment, occlusion of some parts of scenery and texture dependent depth calculations which all leads to erroneous depth reading.

Fig.2: Stereo analysis camera setup.

Depth sensor overcomes from above mentioned draw backs but again it is having downside of low resolution but of course with suitable efficient depth upscaling algorithm in hand this can be surely overcome. Time of Flight sensors and Structured Lighting sensors are most widely used depth sensors .

Time of Flight (ToF) sensor works on ToF principle where the distance measurement is based on the travel time of signal from sensor to object and back to the detector [1],[3]. There are two types, Pulse Runtime ToF sensor where the signal is a pulsed wave and having an accuracy of 10 – 20 mm for few hundred meter depth measurement. And the other one is Continuous Wave ToF sensor where the signal is a continuous wave and having an accuracy of 10 mm for around 10 meter. Fig.3 depicts both kinds of ToF sensors. As Pulse Runtime sensor is having low temporal resolution and hence cannot be used in real time applications. On the other hand Continuous Wave sensor has 60 frames/s temporal resolution and hence most widely used.
Eg. : SR4000 (shown in fig.4), PMD CamCube etc.



(a) (b)

Fig.3: (a) Pulse Runtime ToF Sensor (b) Continuous Wave ToF Sensor.

Fig. 4: SwissRanger SR4000 ToF Sensor.

As mentioned earlier, another most widely used depth sensor is structured lighting sensor. It works on structured lighting principle in which known structure or pattern of light (e.g. grid pattern) is incident on the scenery. The comparison between original light pattern and distorted light pattern allows the depth map construction [2], [3].
Eg. Microsoft Kinect (shown in fig.5).

Fig.5: Microsoft Kinect Structured Lighting Sensor.

Depth map upscaling algorithms are classified into Guided algorithms where ancillary data like color image, intensity image and depth examplars guide the upscaling process and Unguided algorithms where no ancillary data is required for upscaling. This classification is shown in fig.6.Three main approaches come under Guided algorithm.

- Markov Random Field (MRF) approach
- Joint Bilateral Upscaling (JBU) approach
- Edge Weighted Optimization Concept (EWOC) approach

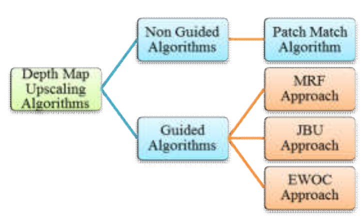


Fig.6: Classification of Depth Map Upscaling Algorithms.

Deibel and Thrun [4] used MRF approach for depth map upscaling in 2006. Low resolution depth data is fused with high resolution texture data to generate high resolution depth map. This is done by using MRF frame work and then by using Conjugate Gradient algorithm MRF is solved. MRF framework has five different kinds of nodes distributed in three layers as shown in fig.7. A node in the figure represents a pixel. First layer representing original depth measurement has less number of nodes. Middle layer representing reconstructed depth map has equal node density as that of third layer which represents guiding color image. This shows the resolution improvement in depth map. Diebel and Thrun are success in filling resolution gap between low resolution depth map and high resolution image by a factor of 10 x. But the issue is blurring of depth map which is overcome in Park *et al.* [5] work.

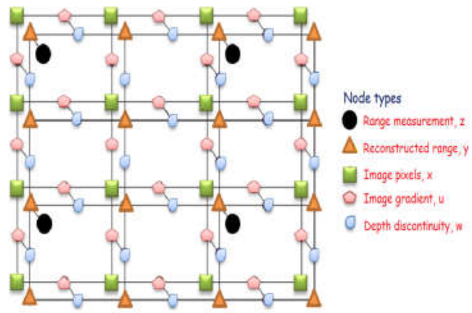


Fig.7: MRF Framework.

In addition to MRF framework, Park has used NLM filtering which preserves useful fine information even in the presence of noisy input data. Blurring of edges in Diebel work is due to the combination of two different depth values at the object boundary. In order to overcome this Park has performed outlier detection for each pixel in the depth map. Outlier detection consists a 9 x 9 window which compares each pixel to local maximum and minimum values. The extent of contrast between these two values indicates the presence of two different depth layers in the window. This method is robust to noise. They have tested the algorithm by adding Gaussian noise to the low resolution depth map and the algorithm succeeded in upscaling even in the presence of noise

JBU filter is an edge preserving filter which consists both spatial filter and range filter and it was first proposed by Tomasi and Manduchi in 1998. Kopf *et al.* [6] used JBU strategy not only for depth map upscaling but also for other image enhancement techniques such as graph cut image operation, tone mapping and colorization in 2007. He has performed the mentioned operations on downsampled image to save computations and memory cost. Then the solution is upsampled using JBU approach. This work has succeeded in getting high resolution depth map with accurate and sharp edges at the object boundaries but has not able to avoid texture copying artefact which is solved in EWOC approach by schwarz.

In the very same year 2007, Yang *et al.* [7] also did work on depth map upscaling by using bilateral filter. Bilateral filter framework is shown in fig.8. He has considered one or two high resolution color images as reference. The original low resolution depth map is upsampled to match color image resolution and it serves as initial depth map, D_0. At first iteration, bilateral filtering is applied to D_0 by color image guidance, generating new depth map D_1. A cost volume, C_1 which indicates deviation of reconstructed depth map D_1 from the actual depth reading is built based on D_0 and D_1. In each iteration depth map refining is done based on minimal cost function. Cost function prevents the deviation of depth reading form the actual reading during refinement process. Here the cost function is discrete in nature which results discontinuous depth map. Hence sub pixel estimation algorithm based on quadratic polynomial interpolation is used to approximate the depth values in the missing regions. This work achieved upscaling factor of 100 x.

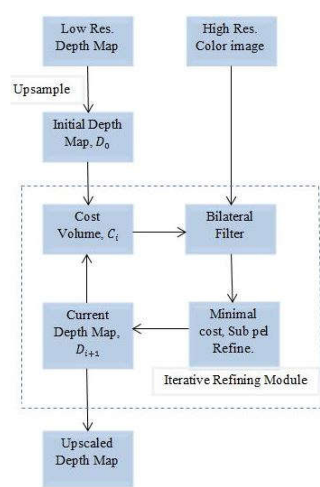


Fig.8: JBU Framework.

In 2014, Schwarz *et al.* [8] introduced a new concept on weighting strategy, EWOC which involved three weights which are applied consecutively as shown in fig.9.

- Edge weight: preserves edge information of video frame.
- Error weight: reduces sensor noise in depth map.
- Temporal weight: preserves temporal consistency of depth values for a pixel in two consecutive frames to avoid flickering in real time video application.

The framework has three inputs: low resolution depth map, its corresponding high resolution video frame and upscaled result from the previous frame. The low resolution depth map, D_L is upscaled to match the high resolution video frame which results in sparse represented depth map, D.

The texture frame, I is passed through an edge filter and then masked with low resolution depth map to generate edge weight, W_E. Active brightness, A from ToF sensor gives error weight, W_A. D(t-1), the previous upscaled depth map and the difference between current and previous texture frame gives temporal weight, W_T. All these weights are used in optimization process to upscale sparse depth map D to dense full resolution depth map. Schwarz succeeded in depth upscaling and specially avoided texture copying artefacts.

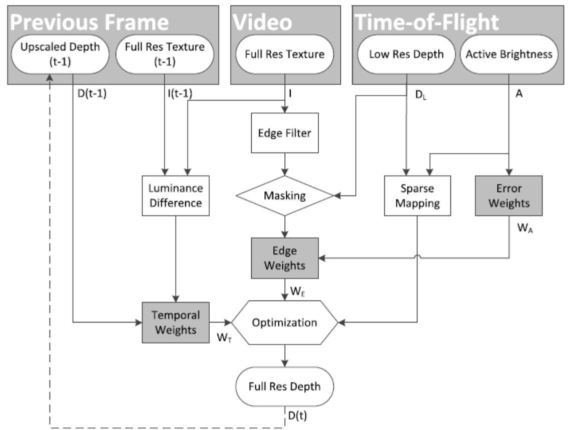


Fig.9: EWOC Framework.

An Unguided approach, 'Patch Match' technique is followed by Hornacek *et al.* [9] for depth map upscaling. This work has not used any ancillary data like color image or intensity image or depth exemplars. It tries to fill the missing values by matched patches within the depth map itself. The patch size is arbitrary based on the scaling factor. This work has evaluated by considering depth data from stereo, ToF, laser scans and structured lighting and was successful in depth map upscaling. Also it undergone Middlebury stereo evaluation and its RMSE score was fairly high compare to other test cases which followed guided approach and this was the only algorithm which succeeded in removing noise then.

III. System Design

The proposed system design is shown in fig.10.

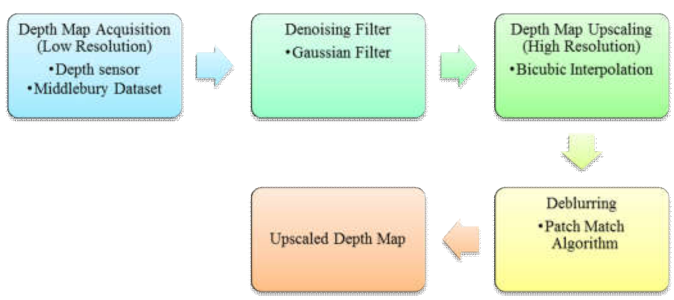


Fig. 10: Block Diagram of the Proposed System.

A. Depth Acquisition

Depth map can be directly captured from depth sensor. Every year Middlebury stereo vision lab uploads depth map dataset for research purpose. One such depth map is considered in this work.

B. Denoising

Depth map captured from depth sensor usually gets affected by Photon shot noise which is caused due to the limited spatial resolution of depth sensor and is approximated as a zero-mean Gaussian, with standard deviation, σ which is defined by the active brightness *A*.

$$\sigma = \frac{1}{A^2} \qquad (1)$$

(1) leads to an assumption that best results are achieved with the highest active brightness. At high active brightness, photo generated electrons flood the capturing pixel element, causing erroneous depth readings. Gaussian filter will be used to remove Photon shot noise. A 2D Gaussian filter is given by (2),

$$g(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{4\sigma^2}} \qquad (2)$$

x and y are the distances from the origin along horizontal and vertical axes respectively. σ is the standard deviation of the Gaussian distribution.

C. Depth Map Upscaling

Upscaling of low resolution depth map is achieved through bicubic interpolation which considers 16 pixels (4x4) for interpolation yielding interpolated surface,

$$p(x,y) = \sum_{i=0}^{3}\sum_{j=0}^{3} a_{ij}x^{i}y^{j} \tag{3}$$

D. Deblurring

Bicubic interpolation results smoothened edges. Hence Patch Extraction algorithm has been used for deblurring whose flow is shown in fig.11.

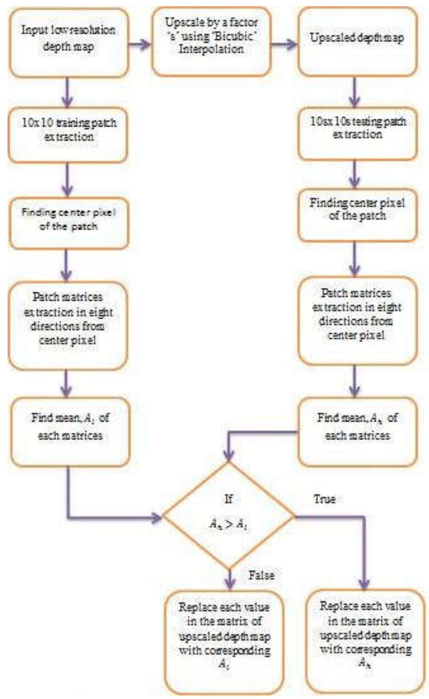


Fig.11: Patch Match algorithm Flow Chart.

IV. SYSTEM IMPLEMENTATION

The system is implemented in MATLAB R2013a. System implementation of Depth map acquisition, Upscaling and Deblurring are explained in the following sections.

A. Depth Map Acquisition

Fig.12 is the block diagram for Depth map Acquisition where the input is a low resolution depth map.

Fig.12: Depth Map Acquisition.

- **Input Image:** Function *uigetfile( )* displays a dialog box to input depth map which specifies the path and file.
- **Image Read:** Using path and file depth map is read using *imread( )*.
- **Image Resize:** The read depth map is resized to the nearest multiples of 10 using *imresize( )*.

Eg. If the input depth map is of 450 x375 then scale it to 400x400.

- **Change Datatype:** An unsigned 8 bit integer image pixel value is converted to double precision i.e 64 bit value using *double( )*.

B. Depth Map Upscaling And Deblurring

Fig.13 is the block diagram for Depth map Upscaling where the input is resized depth map from the previous block.

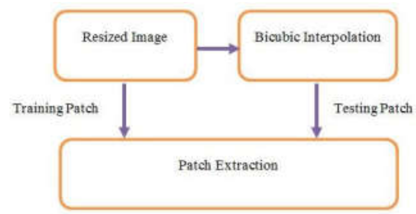


Fig.13: Depth Map Upscaling.

- **Bicubic Interpolation:** The resized input depth map resolution is increased using Bicubic interpolation using *imresize( )* by specifying the input arguments as *'Bicubic'* and scaling factor *'2'*. If the input depth map is of size 100 x 100 then the interpolated depth map is of size 400 x400 as shown in fig.14.

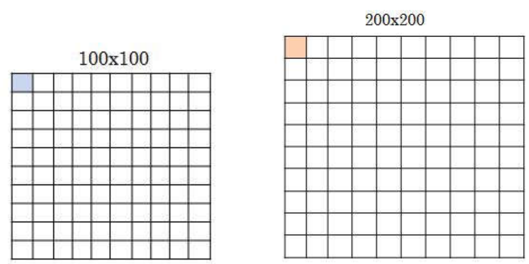


Fig.14: Input 100 x 100 depth map and its corresponding interpolated 200 x200 depth map.

- **Patch Extraction:** 'training patch' and 'testing patch' are extracted from input low resolution and upscaled depth maps respectively.
- ➢ If the training patch size is 10 x10 then testing patch size = scaling factor * training patch size, i.e 20 x 20 as shown in fig.15.

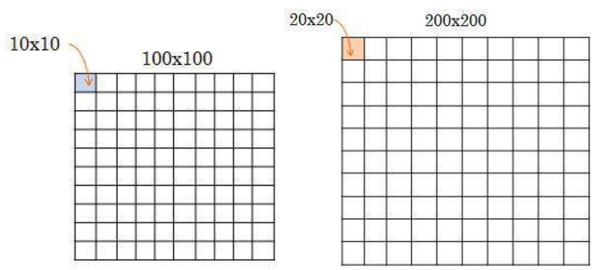


Fig.15:10 x10 patch and its corresponding 20 x20 patch.

- ➢ There are 100 patches in each depth maps which indicate 100 central pixels in each depth maps.
- ➢ Store the central pixel position of each patch of the low resolution depth map in *center(100 x 2)* matrix and the central pixel position of the corresponding patch from upscaled depth map can be obtained by scaling factor * *center(100 x 2).*
- ➢ Extract patches from both the depth maps.
- ➢ Find patch matrices in all the eight directions as shown in fig.16.

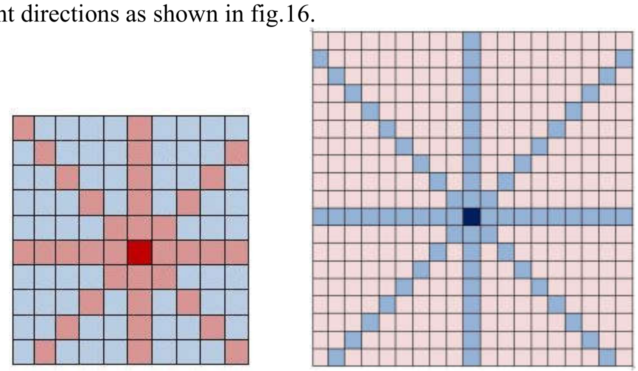


Fig.16: Patch extraction and finding central pixel position and matrices in all the eight directions.

- ➢ Find mean average of the matrices, A_l and A_h. Make comparision and replace the upscaled depth map matrix with greater value.
- ➢ Do the above operations for all the patches.
- ➢ Pseudo codes for central pixels position finding and patch extraction and mean value replacement are given in the fig.17.

➢

```
x = 1;
dh = 1;
for i = 1: h
dw = 1;
    for j = 1: w

      center(x, :) = [k + dh, k + dw ];
      dw = dw + lowPS;
      x = x+1;
      end
dh = dh + lowPS;
end
```

(a)

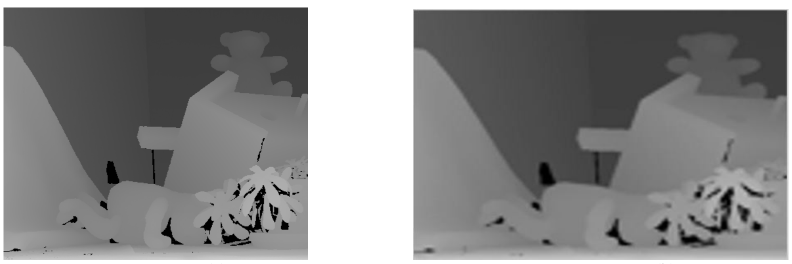
```
for i = 1:nPatch
Patch matrices are extracted in all
nPatch patches from both the
images.
    for j = 1:8

    Find meanAverages A_l and A_h
    from both the images. Make
    comparision and replace
    corresponding patch matrix with
    greater value.
    end
end
```

(b)

Fig.17 (a) Pseudo code for central pixels position finding (b) Pseudo code for patch extraction and meanAverage replacement .

V. Result

An input 400x400 low resolution depth map and its bicubic interpolated 800 x 800 depth map which is smoothened is shown in fig.18.



(a) (b)

Fig.18: (a) Input low resolution 400 x 400 Depth Map (b) 800 x 800 bicubic interpolated depth map.

Sharpened image obtained from patch extraction algorithm is given in fig.19.

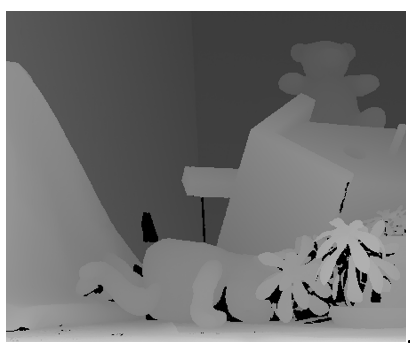


Fig.19: Deblurred high resolution Depth Map.

VI. Application And Advantage

A. Application

Noiseless and upscaled depth maps are used in automatic robotic navigation, 3D TV and entertainment industry, security and surveillance, industrial automation , gaming and space applications.

B. Advantage

The Patch Extraction algorithm is independent of ancillary data. Hence depth map high resolution can be achieved with a single depth sensor even in the absence of color camera in the applications like robotic navigation, security and surveillance and industries. In 3D TV applications even though color camera is available, this independency makes algorithm fast.

VII. CONCLUSION

This work initially considered low resolution depth map. This has been upscaled by a factor of '2' using 'Bicubic interpolation'. The interpolation resulted smooth image. Hence by using 'Patch Extraction' algorithm sharp image has been obtained.

REFERENCES

[1]Miles Hansard, Seungkyu Lee, Ouk Choi, Radu Horaud, "Time-of-Flight Cameras: Principles, Methods and Applications", SPRINGER BRIEFS IN COMUTER SCIENCE, hal-00725654, version 1-7 Dec 2012.

[2]Y.F. Yang and J.K. Aggarwal, "An overview of geometric modeling using active sensing", Control Systems Magazine, IEEE, 8(3): 5–13, June 1988.

[3]Sebastian Schwarz, "Depth Map Upscaling for Three-Dimensional Television: The Edge-Weighted Optimization Concept", Licentiate Thesis No. 92, Sundsvall, Sweden 2012.

[4]J. Diebel, S. Thrun, "An application of markov random fields to range sensing", In Proceedings of Conference on Neural Information Processing Systems, MIT Press, Cambridge, MA, USA, 2005.

[5]J. Park, H. Kim, Y.-W. Tai, M. S. Brown, I. Kweon, " High quality depth map upsampling for 3D-ToF cameras", In Computer Vision (ICCV), 2011, IEEE International Conference, pages 1623–1630, 2011.

[6]J. Kopf, M. F. Cohen, D. Lischinski, M. Uyttendaele, "Joint bilateral upsampling", ACM Transactions on Graphics, 26(3), 2007.

[7]Qingxiong Yang, Ruigang Yang, James Davis, David Nister, "Spatial-Depth Super Resolution for Range Images", IEEE CVPR, Jun, 2007, pp.1-8.

[8]Sebastian Schwarz, Mårten Sjöström, Olsson, "A Weighted Optimization Approach to Time-of-Flight Sensor Fusion", IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 23, NO. 1, JANUARY 2014.

[9]Michael Hornacek, Christoph Rhemann, Margrit Gelautz, and Carsten Rother, "Depth Super Resolution by Rigid Body Self-Similarity in 3D", Microsoft Research Cambridge,Vienna University of Technology.