ALGORITHMS FOR SUPER-RESOLUTION AND RESTORATION OF NOISELESS AND NOISY DEPTH IMAGES

Thesis
Submitted in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

by

CHANDRA SHAKER BALURE

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA
SURATHKAL, MANGALORE - 575025, INDIA

April 2019

DECLARATION

I hereby *declare* that the Research Thesis entitled **Algorithms for Super-Resolutoin and Restoration of Noiseless and Noisy Depth Images** which is being submitted to the **National Institute of Technology Karnataka, Surathkal** in partial fulfillment of the requirements for the award of the Degree of **Doctor of Philosophy** in **Department of Electronics and Communication** is a *bona fide report of the research work carried out by me*. The material contained in this thesis has not been submitted to any University or Institution for the award of any degree.

Chandra Shaker Balure
Register No.: 123017EC12F03
Department of Electronics and Communication Engineering

Place: NITK Surathkal
Date:

CERTIFICATE

This is to *certify* that the Research Thesis entitled **Algorithms for Super-Resolutoin and Restoration of Noiseless and Noisy Depth Images**, submitted by **Chandra Shaker Balure** (Register Number: 123017EC12F03) as the record of the research work carried out by him, is *accepted as a Research Thesis submission* in partial fulfillment of the requirements for the award of degree of **Doctor of Philosophy**.

Ramesh Kini M.
Research Supervisor
Associate Professor
Department Electronics and Communication Engg.
NITK Surathkal - 575025

Chairman - DRPC
(Signature with Date and Seal)

This thesis is dedicated to my parents

Smt. Prabhavati Balure and **Sri. Kantveer Balure**

and my niece

Shreya Yadlapur

ACKNOWLEDGEMENTS

The success and final outcome of this thesis required a lot of guidance and assistance from many people and I am extremely privileged to have got this all along the completion of my thesis. All that I have done is only due to such supervision and assistance and I would not forget to thank them.

I wish to express my foremost thanks of gratitude to my research supervisor Dr. Ramesh Kini M. for his constant support and guidance during my research work. I have been very privileged to have worked under his supervision which has made the completion of the thesis possible. Along with him, I would like to thank other Research Progress Assessment Committee (RPAC) members Dr. Deepu Vijayasenan of Department of Electronics and Communication Engineering and Dr. Jeny Rajan of Department of Computer Science and Engineering for their critic and suggestions to improve the thesis. I would like to thank Dr. T. Laxminidhi, the present Head of the Department of Electronics and Communication Engineering, and the earlier Heads of the Department during my research tenure Prof. U. Shripathi Acharya, Prof. M. S. Bhat and Prof. Murlidhar Kulkarni for the constant support and encouragement

I am thankful to and fortunate enough to get constant encouragement, support and guidance from all the teaching and non-teaching staffs of the department who helped in successfully completing my thesis. I would also like to thank all the fellow research scholars Ranjan, Goutham, Chetan, Sarwesh, Jhnanesh, MANS, Gnani, Shareef, Vasu, Jayaram, Shilpa, Ragesh, Nagraj, Shruti, Prasad, Patil, Mandya and Girish of the department and off the department for holding fruitful and productive discussions inside and outside labs.

I owe my deep gratitude to Dr. Arnav Bhavsar for giving me an opportunity to work with him at Indian Institute of Technology (IIT) Mandi, Himachal Pradesh. I am very much thankful for his technical support and chearful nature, which taught me many things during my internship tenure at IIT Mandi. I also would like to thank Multimedia Analytics System (MAS) lab at IIT Mandi for encouraging to have technical group discussions to expand our knowledge. I would like to thank all professors and research scholars associated with MAS lab for their constant support. I would not forget to mention Srimanta, Pulkit, Vinayak, Anshul, Shaifu, Vibha, Munender, Krishan, Shikha, Krati, Manav, Prabjot, Shristhi, Kartik and Jyothi mam for the memorable trips and parties.

I am grateful to Ministry of Human Resource and Development, Government of India for financially supporting me to carry out this research work. Last but not the least, I would like to thank my parents for their constant support and advice to maintain a proper balance of health and mind. Their never ending support during my research tenure as Ph.D. scholar is invaluable. Finally, I thank God for his blessing and showing me the path to glory.

ABSTRACT

Over the last decade, along with intensity images depth images are also gaining popularity because of its demand in applications like robot navigation, augmented reality, 3DTV, etc. The distinctive characteristic of depth image is that each pixel value represents the distance from the camera position, unlike optical image where each pixel represent intensity values. The prominent features of depth images are the edges and the corners, but it lacks texture unlike optical images. The modern high-end depth cameras provide depth map with higher spatial resolution and higher bit-width, but they are bulky and expensive. However, on the other hand, the commercial low-end depth cameras provide lower spatial resolution, smaller bit width, and are relatively inexpensive. Moreover, the depth images captured by such cameras are noisy and may have some missing regions. To deal with problems like noise and missing regions in the images, the image processing methods like image denoising and image inpainting can be used. Super-resolution (SR) methods address the problem of lower spatial resolution by taking low-resolution (LR) input image and produce high-resolution (HR) image with minimal perturbation in image details.

In literature, several super-resolution (SR) and depth reconstruction (DR) methods have been proposed to address the problems associated with these low-end depth camera. We propose few methods to address the above mentioned issues related to the process of super resolution and restoration of depth images.

Wavelets have been used for decades for image compression, image denoising and image enhancement because of its better localization in time (space) and frequency. In the proposed work, a wavelet transform based single depth image SR method has been proposed. It uses discrete wavelet transform (DWT), stationary wavelet transform (SWT), and the image gradient. The proposed method is an intermediate stage for obtaining the high-frequency contents from different subbands obtained through DWT,

SWT and gradient operations on the input LR image and estimates the SR image.

For super-resolution by larger factors, i.e. $\times 4$ or $\times 8$ or higher, the guided approach has been used in literature which makes use of the corresponding HR guidance colour image which are easy to capture. In this work, we propose a HR colour-image guided depth image SR method that makes use of the segment cues from the HR colour image. The cues are obtained by segmentation of the HR colour image using popular segmentation methods such as mean-shift algorithm (MS) or simple linear iterative clustering (SLIC) segmentation algorithms. Like other guidance image based methods, it is assumed that the prominent edges in the depth image coincides with the edges in the HR guidance colour image. The median of a segment in the initial estimated depth image corresponding to the segment in the guiding HR colour image is computed. This median value replaces the depth value in that identified segment of the initial estimated depth image. After processing all the segments, we get a final SR output with better edge details and reduced noise. Bilateral filtering can be applied as post processing to smooth the variations at the abutting segment regions. The initial estimate of the SR depth image is derived from LR depth image using the following two approaches. The first one is with bicubic interpolation to the required spatial resolution and the SR process which uses this is referred as LRBicSR method in this work. The other method maps the LR points on to the HR grid and super resolves; this method is referred to as LRSR method.

Processing of sparse depth images involves two stages namely DR and SR in that order. This framework of DR followed by SR is called as DRSR method and is challenging. The sparse depth images used for processing may have sparseness range between 1% and 15% of the total pixels. Processing of very sparse depth images of the order of 1% is highly challenging and has been reasonably reconstructed. The corresponding RGB images have been used for guiding the reconstruction process. Two approaches have been proposed to estimate the unknown depth values in the sparse depth input. First one being the plane fitting approach (PFit) and the other being the median filling approach (MFill). This work also shows that guidance based methods are useful in overcoming the effect of noise in depth images and inpainting of the missing regions in the depth images.

Literature contains SR methods for intensity images that use a set of training images to learn the HR-LR relationship. In this work, a learning based method has been proposed where algorithm learns the image details from the HR and LR pairs of training images using Gaussian mixture model (GMM). It has been observed from the conducted experiments that, for larger SR factors, the learned parameters do not help much in learning the finer details. So, hierarchical approach has been proposed for such factors and the approach tend to give better SR image quality.

The anisotropic total generalized variation method (ATGV) available in the literature is an iterative method and the quality of the SR image so obtained with this method is dependent on the number of iterations used. A simple and less computationally intensive Residual interpolation method (RI) has been used as a preprocessor for ATGV. The computational complexity of RI is comparable to the computational intensity of classical bicubic interpolation method. RI provides a better initial estimate to the ATGV. It has been observed that the proposal of cascading the RI as a preprocessor reduces the number of iterations, converges faster to achieve the better SR image quality.

For experimentation, we have used the freely available *Middlebury* depth dataset, which has depth images along with their corresponding registered colour image. Another dataset used is *Kitti* dataset which has depth images of outdoor scenes. Real-time depth images captured from Kinect camera and ToF camera has also been used in the experiments to show the robustness of the proposed methods. The LR image is generated from the ground truth (GT) image by blurring, downsampling and adding noise to it. Several performance metrics e.g. peak signal-to-noise ratio (PSNR), structural similarity (SSIM) and root mean square error (RMSE) have been used to evaluate the performance.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS **v**

ABSTRACT **vii**

LIST OF FIGURES **xx**

LIST OF TABLES **xxiii**

ABBREVIATIONS **xxiv**

1 INTRODUCTION **1**

1.1 BRIEF DESCRIPTION OF SUPER-RESOLUTION 1
1.2 SUPER-RESOLUTION PREVIEW 9
1.3 IMAGE MODELING 11
1.4 MOTIVATION AND CHALLENGES 13
1.5 RESEARCH OBJECTIVES 15
1.6 IMAGE DATASET 16
1.7 PERFORMANCE METRICS 17
1.8 THESIS CONTRIBUTION 18
1.9 THESIS ORGANIZATION 19

2 LITERATURE REVIEW **25**

2.1 SINGLE DEPTH IMAGE SR METHODS 25
2.2 MULTIPLE IMAGE DEPTH SR METHODS 27
2.3 GUIDANCE BASED DEPTH IMAGE SR METHODS 29
2.4 TRAINING BASED DEPTH SR METHODS 31
2.5 DEPTH RESTORATION FROM NON-UNIFORM SAMPLES . . . 32

3 WAVELET TRANSFORM BASED SINGLE DEPTH IMAGE SUPER-RESOLUTION 35

3.1 INTRODUCTION 35

3.2 PROPOSED METHOD FOR WAVELET BASED DEPTH IMAGE SR 38

3.2.1 Initial Estimate 38

3.2.2 High Frequency Combination 40

3.2.3 HR Image Estimation 42

3.2.4 Bilateral Filtering 43

3.3 EXPERIMENTAL RESULTS AND DISCUSSIONS 44

3.4 SUMMARY 46

4 DEPTH IMAGE SUPER-RESOLUTION USING HR GUIDANCE COLOUR IMAGE 53

4.1 INTRODUCTION 53

4.2 SEGMENTATION ALGORITHMS 56

4.3 PROPOSED HR GUIDANCE IMAGE BASED SR METHOD *WITH* INITIAL BICUBIC ESTIMATE 59

4.3.1 Initial Estimate 60

4.3.2 HR Colour Image Segmentation 60

4.3.3 SR Image Estimation 61

4.3.4 Bilateral Filtering 63

4.4 PROPOSED SR GUIDANCE BASED SR METHOD *WITHOUT* INITIAL ESTIMATE 64

4.5 EXPERIMENTAL RESULTS AND DISCUSSIONS 67

4.5.1 Results of LRBicSR Method 67

4.5.2 Results of LRSR Method 71

4.5.3 Results of LRSR Method on Kitti Dataset 78

4.6 SUMMARY 78

5 RECONSTRUCTION AND SUPER-RESOLUTION OF SPARSE DEPTH IMAGE 81

5.1 INTRODUCTION 81

5.2 RGB GUIDANCE IMAGE BASED DEPTH RECONSTRUCTION FROM SPARSE DATA 86
  5.2.1 Depth Reconstruction by Plane Fitting 86
  5.2.2 Depth Reconstruction by Median Filling 88
5.3 RGB GUIDANCE IMAGE BASED DEPTH IMAGE SR FROM SPARSE LR INPUT 91
5.4 RELEVANT APPLICATIONS BASED ON PROPOSED APPROACH 94
  5.4.1 Depth Denoising 94
  5.4.2 Depth Inpainting 95
5.5 EXPERIMENTAL RESULTS AND DISCUSSIONS 96
  5.5.1 Results of DR 96
  5.5.2 Results of DRSR 101
  5.5.3 Results of Depth Image Denoising 106
  5.5.4 Results of Depth Image Inpainting 108
  5.5.5 Failure Scenario 110
5.6 SUMMARY 111

6 GAUSSIAN MIXTURE MODEL BASED SINGLE DEPTH IMAGE SUPER-RESOLUTION 113
6.1 INTRODUCTION 113
6.2 BACKGROUND 114
  6.2.1 Gaussian Mixture Model Description 115
  6.2.2 Expectation-Maximization Algorithm Description 117
6.3 PROPOSED LEARNING BASED GMM MODEL FOR DEPTH IMAGE SR 118
  6.3.1 SR GMM Training 119
  6.3.2 Multivariate GMM Testing 121
  6.3.3 Direct Approach versus Hierarchical Approach 123
6.4 EXPERIMENTAL RESULTS AND DISCUSSIONS 124
  6.4.1 Training/Testing Database 125
  6.4.2 Parameter Selection 125

6.4.3 SR Results on Synthetic Depth Images 126
6.4.4 SR Results From Direct and Hierarchical Approach 138
6.4.5 SR Results on Real Depth Images 144
6.4.6 Time Complexity 145
6.5 SUMMARY 147

7 BETTER INITIAL ESTIMATE FOR ITERATIVE SUPER-RESOLUTION METHODS 149
7.1 INTRODUCTION 150
7.2 BACKGROUND DETAILS 151
7.2.1 Brief Description of Residual Interpolation Method 151
7.2.2 Brief Description of Anisotropic Total Generalized Variation Method 154
7.3 PROPOSED METHOD 157
7.4 EXPERIMENTAL RESULTS AND DISCUSSIONS 158
7.5 SUMMARY 163

8 CONCLUSION AND FUTURE DIRECTIONS 165
8.1 CONCLUSION 165
8.2 FUTURE DIRECTIONS 167

A INTERPOLATION METHODS 169

REFERENCES 173

LIST OF PAPERS BASED ON THESIS 181

CURRICULUM VITAE 183

LIST OF FIGURES

1.1 Time-of-flight (ToF) depth cameras 3

1.2 Depth image in two different display format 4

1.3 Image acquisition system 4

1.4 Increasing spatial resolution of intensity image 7

1.5 Increasing spatial resolution of depth image 7

1.6 Camera sensors of equal size with different number of CCD elements. Resolution of sensors from L to R: 8×8, 16×16, 24×24 8

1.7 Idea of super-resolution with input as LR image and output as HR image. 8

1.8 Forward and inverse process for HR-LR image 9

3.1 Wavelet transformation 36

3.2 Block diagram of the proposed wavelet based method for single depth image super-resolution by factor α. The subbands A, H, V, D and a, h, v, d are the outcomes of one level SWT and DWT operation respectively. The subbands aI, hI, vI, dI are the interpolated versions of DWT outcomes, f_x, f_y are the gradients, and aF, hF, vF, dF are the final subbands for IDWT process. 39

3.3 One level DWT operation on an image with low-pass (LP) and high-pass (HP) filter banks to produce four subbands. 41

3.4 SR results comparison for upsampling factor $\times 4$ on *noiseless* ($\sigma = 0$) depth images *Cones*. **Top row**: Ground truth, DWT (Demirel and Anbarjafari, 2011a), SWT (Demirel and Anbarjafari, 2011b), Proposed. **Bottom row**: zoomed region of the above images respectively. . . . 47

3.5 SR results comparison for upsampling factor $\times 4$ on *noisy* ($\sigma = 5$) depth images *Cones*. **Top row**: Ground truth, DWT (Demirel and Anbarjafari, 2011a), SWT (Demirel and Anbarjafari, 2011b), Proposed. **Bottom row**: zoomed region of the above images respectively 47

3.6 SR results comparison for upsampling factor $\times 4$ on *noiseless* ($\sigma = 0$) depth images *Art*. **Top row**: Ground truth, DWT (Demirel and Anbarjafari, 2011a), SWT (Demirel and Anbarjafari, 2011b), Proposed. **Bottom row**: zoomed region of the above images respectively. . . . 48

3.7 SR results comparison for upsampling factor ×4 on *noisy* ($\sigma = 5$) depth images *Art*. **Top row**: Ground truth, DWT (Demirel and Anbarjafari, 2011a), SWT (Demirel and Anbarjafari, 2011b), Proposed. **Bottom row**: zoomed region of the above images respectively 48

3.8 SR results comparison for upsampling factor ×4 on *noiseless* ($\sigma = 0$) depth images *Reindeer*. **Top row**: Ground truth, DWT (Demirel and Anbarjafari, 2011a), SWT (Demirel and Anbarjafari, 2011b), Proposed. **Bottom row**: zoomed region of the above images respectively . . . 49

3.9 SR results comparison for upsampling factor ×4 on *noisy* ($\sigma = 5$) depth images *Reindeer*. **Top row**: Ground truth, DWT (Demirel and Anbarjafari, 2011a), SWT (Demirel and Anbarjafari, 2011b), Proposed. **Bottom row**: zoomed region of the above images respectively 49

3.10 Average PSNR result comparison of proposed SRDWT method with other SR methods for several upsampling factors on both *noiseless* (**left**) and *noisy* (**right**) images. Notation $\times i\_nj\_basis$ indicates upsampling factor i, noise standard deviation j, and wavelet basis function $basis$. 50

4.1 Depth image upsampling aided by the HR color image. 54

4.2 MS segmentation results on *cones* colour image 58

4.3 SLIC segmentation results on *cones* colour image 59

4.4 Block diagram of the proposed approach for depth image super-resolution. 60

4.5 Block diagram of the proposed LRSR method using guidance colour image. 65

4.6 Direct approach of LRSR from LR image mapped uniformly on the HR grid. 66

4.7 Hierarchical approach of LRSR from input LR image mapped uniformly on the HR grid of double size. 66

4.8 SR output comparison of proposed LRBicSR method with other SR methods for upsampling factor ×4 on *noiseless* image *Cones* with zoomed region inset at bottom-right corner. **From left**: GT, Bicubic, GIF, Proposed (SR-MS-BF) 69

4.9 SR output comparison of proposed LRBicSR method with other SR methods for upsampling factor ×4 on *noisy* image *Cones* with zoomed region inset at bottom-right corner. **From left**: GT, Bicubic, GIF, Proposed (SR-MS-BF) 69

4.10 SR output comparison of proposed LRBicSR method with other SR methods for upsampling factor ×8 on *noiseless* image *Reindeer* with zoomed region inset at bottom-right corner. **From left**: GT, Bicubic, GIF, Proposed (SR-MS-BF) 70

4.11 SR output comparison of proposed LRBicSR method with other SR methods for upsampling factor $\times 8$ on *noisy* image *Reindeer* with zoomed region inset at bottom-right corner. **From left**: GT, Bicubic, GIF, Proposed (SR-MS-BF) 70

4.12 SR output comparison of proposed LRBicSR method with other SR methods for upsampling factor $\times 8$ on *noisy* image *Art* with zoomed region inset at bottom-right corner. **From left**: GT, Bicubic, GIF, Proposed (SR-MS-BF) 70

4.13 Average PSNR result comparison of proposed LRBicSR method with other SR methods for upsampling factor $\times 4$ and $\times 8$ on both *noiseless* and *noisy* images. 73

4.14 SR output comparison of proposed LRSR method and its variants with other SR methods for upsampling factor $\times 2$ using MS and SLIC segment cues along with their cropped regions. 75

4.15 SR output comparison of proposed LRSR method and its variants with other SR methods for upsampling factor $\times 4$ using MS segment cues by direct and hierarchical approach along with their cropped regions. . 76

4.16 SR output comparison of the variant of proposed LRSR method on one of the image from Kitti dataset for $\times 2$ upsampling factor. The images shown here are adjusted for dynamic range of the depth only for the display purpose, however, the original images are darker. **From Top**: Bicubic, SR-Dir-MFill, GT. 79

5.1 Depth reconstruction from sparse random depth pixels 82

5.2 Depth reconstruction from 1% random depth samples. **Left to right**: 1% sparse depth data points, PFitDR, MFillDR, GT 83

5.3 Block diagram of DR-PFit method 86

5.4 Block diagram of proposed DR-MFill method. 89

5.5 Output of proposed two-stage DR-MFill method for decreasing samples (L to R: 50%, 40%, 10%, 5% and 1% visible pixel). **Top row**: stage-1 output, and the effect on the number of empty segments. **Bottom row**: stage-2 output, and their corresponding PSNR and SSIM values. . . 90

5.6 Super-resolution from sparse depth points 92

5.7 Depth reconstruction results of image *cones* from only 10% visible random depth pixels. **Top-row**: Sparse depth input, GT, ADMM, DR-DRU; **Bottom-row**: DR-PFit-MS, DR-MFill-MS, DR-PFit-SLIC, DR-MFill-SLIC 98

5.8 Depth reconstruction results of image *cones* from only 5% visible random depth pixels. **Top-row**: Sparse depth input, GT, ADMM, DR-DRU; **Bottom-row**: DR-PFit-MS, DR-MFill-MS, DR-PFit-SLIC, DR-MFill-SLIC 98

5.9 Depth reconstruction results of image *cones* from only 1% visible random depth pixels. **Top-row**: Sparse depth input, GT, ADMM, DR-DRU; **Bottom-row**: DR-PFit-MS, DR-MFill-MS, DR-PFit-SLIC, DR-MFill-SLIC 99

5.10 DRSR results for upsampling factor $\times 2$ and $\times 4$ from sparse LR image with 50% visible depth pixels 104

5.11 Cropped region of images from Figure 5.10 104

5.12 DRSR results for upsampling factor $\times 2$ and $\times 4$ from sparse LR image with 10% visible depth pixels 104

5.13 Cropped region of images from Figure 5.12 104

5.14 Results of depth denoising from noisy images with different noise standard deviation (i.e. $\sigma = 5, 10$) 108

5.15 Results of depth inpaiting from random missing regions, structural missing regions and real time Kinect images 109

5.16 Results of depth inpaiting from random scribbled image 110

5.17 Failure depth super-resolution (*Bowling1* image). Poor performance at segment regions with low-contrast (*red* box), and good performance at segment regions with high-contrast (*blue* box) region 111

6.1 Few examples of synthetic training images 125

6.2 Visual comparison of SR results upsampled by $\times 2$ factor on noiseless image. 128

6.3 Visual comparison of SR results upsampled by $\times 2$ factor on noisy image. 129

6.4 Visual comparison of SR results upsampled by $\times 4$ factor on noiseless image. 129

6.5 Visual comparison of SR results upsampled by $\times 8$ factor on noiseless image. 130

6.6 Comparison of average PSNR values of depth super-resolution methods with different number of Gaussian mixtures for upsampling factors $\times 2$, $\times 4$ and $\times 8$ on *noiseless* and *noisy* images. 133

6.7 Plot of average PSNRs of SRGMM method with different Gaussian mixtures with other various depth image SR methods on *one-third* resolution and *full* resolution *noiseless* and *noisy* images. 136

6.8 Comparison of average PSNR values of depth super-resolution by factor $\times 2$ with different HR-LR patch sizes. 138

6.9 SRGMM performance analysis with different number of training patches for upsampling factor $\times 2$, with fixed HR-LR patch size (i.e. 8-4). (a) on noiseless images, (b) on noisy images. 138

6.10 Qualitative results comparison for SR by factor $\times 4\_n0$ (Image: *Art*) 140

6.11 Qualitative results comparison for SR by factor $\times 8\_n0$ (Image: *Baby*) 141

6.12 Qualitative results comparison for SR by factor $\times 4\_n5$ (Image: *Art*) 142

6.13 Qualitative results comparison for SR by factor $\times 8\_n5$ (Image: *Baby*) 143

6.14 SR results of SRGMM on real tof depth images. 145

6.15 Cropped region of SR results of SRGMM on real tof depth images. . 146

7.1 Block diagram of residual interpolation method Konno et al. (2015) 152

7.2 Block diagram of anisotropic total generalized variation method Ferstl et al. (2013) 154

7.3 Block diagram of proposed method combining RI and ATGV in cascade form. 157

7.4 SR results comparison of *noisy* ($\sigma = 5$) depth image *Cones*. **Col1**: SR by factor $\times 2$, **Col2**: SR by factor $\times 4$, **Col3**: SR by factor $\times 8$. 160

7.5 SR results comparison of *noisy* ($\sigma = 5$) depth image *Art*. **Col1**: SR by factor $\times 2$, **Col2**: SR by factor $\times 4$, **Col3**: SR by factor $\times 8$. 161

7.6 SR results of the proposed method under different noise levels for upsampling factors $\times 2$. **Row1**: outputs from proposed method, **Row2**: cropped and zoomed region of their corresponding top images. . . . 162

7.7 SR results of the proposed method under different noise levels for upsampling factors $\times 4$. **Row1**: outputs from proposed method, **Row2**: cropped and zoomed region of their corresponding top images. . . . 162

7.8 SR results of the proposed method under different noise levels for upsampling factors $\times 8$. **Row1**: outputs from proposed method, **Row2**: cropped and zoomed region of their corresponding top images. . . . 163

A.1 Nearest Neighbor Interpolation 169

A.2 Bilinear Interpolation 170

A.3 Bicubic Interpolation 170
A.4 Lanczos kernel for a=3 171

LIST OF TABLES

3.1 PSNR/SSIM comparison of SR by factor 2, 4 and 8 on *noiseless* depth images ($\sigma = 0$) 51

3.2 PSNR/SSIM comparison of SR by factor 2, 4 and 8 on *noisy* depth images ($\sigma = 5$) 51

4.1 Comparison of PSNR/SSIM quantitative result of the proposed LR-BicSR method using MS and SLIC segment cues without and with BF filter on *noiseless* and *noisy* images with other SR methods for $\times 4$ upsampling factor 71

4.2 Comparison of PSNR/SSIM quantitative result of the proposed LR-BicSR method using MS and SLIC segment cues without and with BF filter on *noiseless* and *noisy* images with other SR methods for $\times 8$ upsampling factor 72

4.3 Comparison of PSNR/SSIM quantitative result of the proposed LRSR method for upsampling factor $\times 2$, $\times 4$ and $\times 8$ on depth images without and with noise. Notation $\times i\_\text{sig}j$ indicates SR for upsampling factor i on images with noise standard deviation j. **First best results in bold** 77

4.4 Average PSNR/SSIM results of proposed LRSR method for different upsampling factors on Kitti dataset 78

5.1 PSNR/SSIM results of depth reconstruction (DR) from 10%, 5% and 1% visible data using MS and SLIC segment cues. Best results are in **bold** 100

5.2 Parameters used in the proposed method 101

5.3 PSNR/SSIM results of DRSR for upsampling factor $\times 2$ and $\times 4$ from sparse LR image with 50% visible depth point using MS and SLIC segment cues with MFill approach. Notation $\times i\_\text{sig}j$ indicates SR for upsampling factor i on images with noise standard deviation j. **First best results in bold** 105

5.4 PSNR/SSIM results of DRSR for upsampling factor $\times 2$ and $\times 4$ from sparse LR image with 10% visible depth point using MS and SLIC segment cues with MFill approach. Notation $\times i\_\text{sig}j$ indicates SR for upsampling factor i on images with noise standard deviation j. **First best results in bold** 106

5.5 PSNR/SSIM results of DRSR for upsampling factor $\times 2$ and $\times 4$ from sparse LR image with 5% visible depth point using MS and SLIC segment cues with MFill approach. Notation $\times i$_sig$j$ indicates SR for upsampling factor i on images with noise standard deviation j. **First best results in bold** 106

5.6 PSNR/SSIM results of DRSR for upsampling factor $\times 2$ and $\times 4$ from sparse LR image with 50%, 10% and 5% visible depth point using MS segment cues with PFit approach. Notation $\times i$_sig$j$ indicates SR for upsampling factor i on images with noise standard deviation j. **First best results in bold** 107

5.7 PSNR/SSIM results of depth denoising 108

5.8 PSNR/SSIM results of inpainting method on random scribbled images 110

6.1 Selection of patch sizes for different upsampling factor 126

6.2 PSNR/SSIM performance metrics for SRGMM for upsampling factor $\times 2$ (**Bold** represents the best result among comparative methods) . . 131

6.3 PSNR/SSIM performance metrics for SRGMM for upsampling factor $\times 4$ (**Bold** represents the best result among comparative methods) . . 131

6.4 PSNR/SSIM performance metrics for SRGMM for upsampling factor $\times 8$ (**Bold** represents the best result among comparative methods) . . 132

6.5 Average PSNRs of SRGMM method with different Gaussian Mixtures for single depth image SR problem (**Bold** represents the best result among comparative methods) 132

6.6 Comparison of average PSNRs of SRGMM method with different Gaussian mixtures with other various depth image SR methods on one-third resolution and full resolution *noiseless* and *noisy* images (**Bold** represents the best result among comparative methods) 135

6.7 Average PSNRs of SRGMM method for upsampling factor $\times 2$, with different Gaussian Mixtures, for single depth image SR problem (**Bold** represents the best result among comparative methods) 137

6.8 PSNR/SSIM result comparison of SR for factor $\times 2$, $\times 4$ and $\times 8$ on *noiseless* images (Red text indicate highest value and blue text indicate second highest value) 141

6.9 PSNR/SSIM result comparison of SR for factor $\times 2$, $\times 4$ and $\times 8$ on *noisy* images (Red text indicate highest value and blue text indicate second highest value) 143

6.10 Time complexity for SR Factor $\times 2$ (in sec.) 147

7.1 MSE results of depth super-resolution by factor $\times 2$, $\times 4$ and $\times 8$ for lowest and highest noise levels, i.e. $\sigma = 1$ and $\sigma = 5$ 163

7.2 MSE results of depth super-resolution by factor $\times 2$, $\times 4$ and $\times 8$ for different noise levels, i.e. $\sigma = 2, 3$ and 4 164

ABBREVIATIONS

2D	2-Dimension
2.5d	2.5-Dimension
3D	3-Dimension
CPU	Centeral Processing Unit
ADC	Analog-to-Digital Converter
ATGV	Anisotropic Total Generalized Variation
ADMM	Alternating Direction Method of Multipliers
Bil	Bilnear
Bic	Bicubic
BF	Bilateral Filtering
CM	Credibility Map
CCD	Charge-Coupled Devices
CMOS	Complementary Metal Oxide Semiconductor
CPU	Central Processing Unit
CNN	Convolutional Neural Network
CWT	Continuous Wavelet Transform
DR	Depth Reconstruction
DWT	Discrete Wavelet Transform
DISR	Depth Image Super-Resolution
DRU	Depth Restoration from Undersampled data
dB	Decibel
Dir	Direct
EM	Expectation Maximization
FT	Fourier Transform
FPS	frames-per-second

GMM	Gaussian Mixture Model
GT	Ground Truth
GIF	Guided Image Filtering
GB	Giga Byte
GHz	Giga Hertz
GPU	Graphical Processing Unit
HD	High Definition
Hier	Hierarchical
HMI	Human-Machine Interaction
HP	High Pass
HR	High-Resolution
HL	High-Low
HH	High-High
IDWT	Inverse Discrete Wavelet Transform
IR	Infrared
ISWT	Inverse Sationary Wavelet Transform
JBU	Joint Bilateral Upsampling
JBM	Joint Bilateral Weighted Median
LP	Low Pass
LR	Low-Resolution
LL	Low-Low
LH	Low-High
L	Lakh
MP	MegaPixel
MFill	Median Filling
MISR	Multiple Image Super-Resolution
MS	Mean-Shift
MRF	Markov Random Field
ML	Maximum Likelihood
MSE	Mean Square Error

MMSE	Minimum Mean Square Error
NN	Nearest Neighbor
NEDI	New Edge Directed Interpolation
NAFDU	Noise Aware Filtering for Depth Upsampling
NI-LBP	Neighbor Intensity of Local Binary Pattern
OS	Operating System
PSNR	Peak Signal-to-Noise-Ration
PMD	Photonic Mixer Device
PFit	Plane Fitting
PWAS	Pixel Weighted Average Strategy
PWASI	Pixel Weighted Average Strategy with Intensity image
PWASD	Pixel Weighted Average Strategy with Depth image
POCS	Projection On Convex Set
RMSE	Root Mean Square Error
RGB	Red-Green-Blue
RGB-D	Red Green Blue - Depth
RANSAC	Random Sample Consensus
RV	Random Variable
RAM	Random Access Memory
RI	Residual Interpolation
SR	Super-Resolution
SSIM	Structural Similarity
SWT	Sationary Wavelet Transform
SISR	Single Image Super-Resolution
SLIC	Simple Linear Iterative Clustering
SCN-MSE	Subtraction of Center from Neighbors - Mean Square Error
STFT	Short Time Fourier Transform
TV	Television
ToF	Time-of-Flight
UML	Unifiled Multi-Lateral

UBM	Universal Background Model
VLSI	Very Large Scale Integration
VGA	Video Graphics Array
WT	Wavelet Transform
WZP-CS	Wavelet Zero Padding and Cycle Spinning

CHAPTER 1

INTRODUCTION

1.1 BRIEF DESCRIPTION OF SUPER-RESOLUTION

Digital images are growing tremendously. Capturing digital images have become very easy with just a click of a button. Digital cameras have evolved over past few years from a heavy and bulky cameras to a very compact and portable devices. The modern optical cameras are capable of capturing intensity images at a very higher resolution. The image capturing mechanism of such cameras are based on the principal of optics where light rays from the light source falls on the object and the reflected rays are captured by the camera sensor. The charge-coupled device / complementary metal-oxide semiconductor (CCD/CMOS) image sensor decodes the received reflected analog rays into digital pixel values using analog-to-digital (ADC) converters to estimate the intensity values at all the pixel location. The images are assumed to be captured from a pin-hole camera. Such an imaging system is thought of as a transformation from 3D world to a 2D image.

In recent years, the high resolution display devices have also grown exponentially from VGA to HD-720 to HD-1080 and more. To deal with such devices one need to have high-resolution (HR) acquisition system which meets the display requirement. The resolution of image has become an important aspect now. Along with optical images, the depth images are also becoming popular because of its huge demand in real-time applications like robot navigation, human machine interaction (HMI), automotive driver assistant, gesture interfaces, deictic references in augmented reality, 3D modeling, 3D-TV and many more. In spite of the huge demand for depth images the depth cameras are not able to reach the potential capabilities offered by the modern optical cameras in terms of the spatial resolution and other functionalities.

Resolution

Resolution is the capability to observe the smallest object with distinct boundaries. In general, resolution refers to the number of pixels on a sensor plate. It can be broadly classified into pixel resolution, spatial resolution, spectral resolution, temporal resolution and radiometric resolution. *Pixel resolution* is defined as the number of active pixels on an image sensor. The pixel resolution is more if there are more number of active pixels on an image sensor plate. An image with 640 pixels in width and 480 pixels in height has total of 640x480 = 307200 pixels (0.3 megapixels). *Spatial resolution* is defined as the ability to resolve lines closely placed. A low spatial resolution image will not be able to differentiate between two objects placed relatively close together as compared to a high spatial resolution image. *Spectral resolution* is the ability to resolve spectral features and bands into their separate components. *Temporal resolution* refers to the ability to distinguish the events at different points in time. A video of 1 second time with 30 frames gives less distinction of events as compared to 300 frames. *Radiometric resolution* determines how finely a system can distinguish between intensity levels. An 8-bit system can distinguish 256 intensity levels. This thesis will be focusing only the spatial resolution unless otherwise specified.

Depth Image

Depth images have special property which makes it distinct from the optical images. In depth images, each pixel represent the position of an object from the camera position. The modern depth cameras available in market, some of which are shown in Figure 1.1 are the cameras which capture depth images based on the principle of time-of-flight (ToF).

The depth cameras have different hardware setup to capture the depth image. It does not rely on the optical source for illuminating the scene, instead it uses infrared (IR) signal to estimate the depth of the object. Typical wavelength range of IR signal is from $1\mu m$ to $10^3\mu m$. The depth cameras work on the principle of ToF. These cameras have

(a) SwissRanger 4000 (by MESA Imaging (b) CamCube (by PMD Technologies) (c) Kinect (by Microsoft)

Figure 1.1: Time-of-flight (ToF) depth cameras

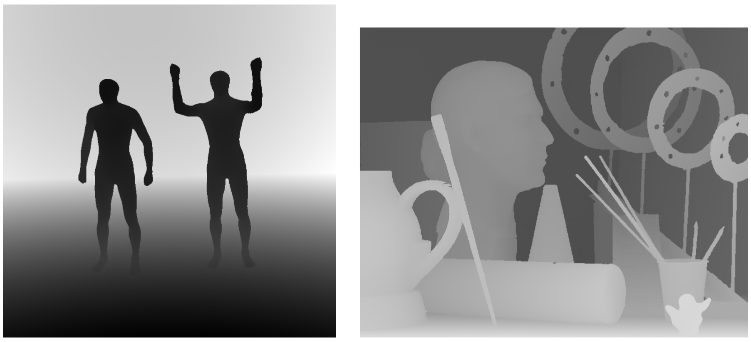
an IR projector and an IR sensor, where the IR projector sends an IR pulse and the IR sensor receives the reflected pulse from the object. The time difference between sending and receiving IR pulse announce the depth of the object from the camera position. For instance, the depth of the scene (in meters) from the camera is given by Eq. 1.1,

$$D_{max} = \frac{c}{2} . \frac{\Delta\phi}{2\pi f} \tag{1.1}$$

where, c is the speed of the light (2.9×10^8 m/s), $\Delta\phi$ is the phase angle between the transmitted and received signal, and f is the frequency of the signal.

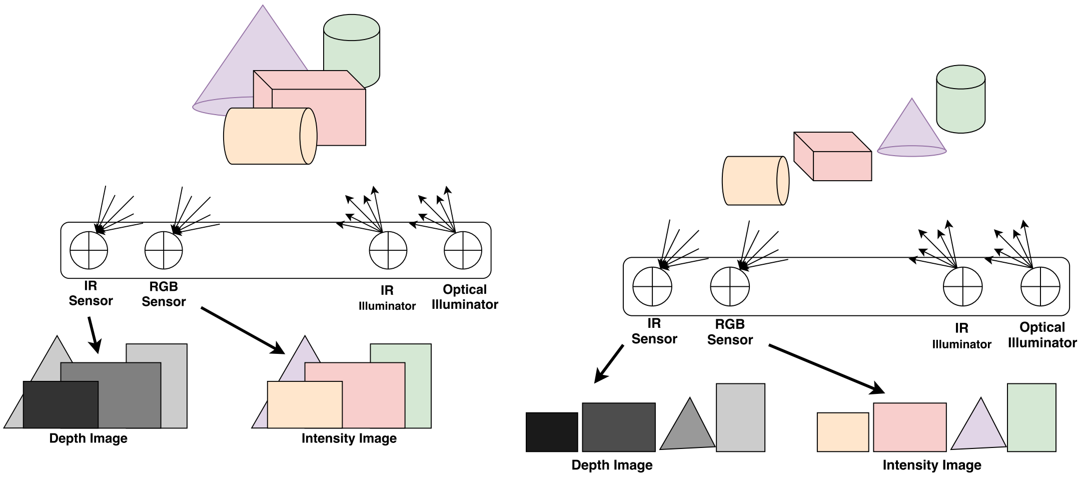
From the principle of ToF for depth imaging, the time difference between the emitted pulse and received pulse denote the distance of the objects in the scene from the camera position, as shown by Eq. 1.1. Hence, every pixel with valid value represent the object distance. The real-time depth images gets affected by the noise. To remove the noise from the image, we must know the type of the noise and its associated parameters like mean and variance. From our experiments, we observed that the probability density distribution (pdf) plot of a planer patch follows Gaussian distribution than the Rayleigh or any other distribution.

Depth images can be displayed in two ways. One is, the closer objects are displayed with darker shades of gray level, and farther objects are displayed with brighter shades (called display-1 here). It is based on the the actual distance of the object from the camera position. The other way of display is opposite to the earlier one, where, the closer objects are displayed with brighter shades and farther objects are displayed with darker shades (called display-2 here). It is based on the parallax effect where the near objects displace more and the far objects displace less when the viewing position changes from either left-to-right or from right-to-left. Example of both the depth images are shown in Figure 1.2.



(a) Display-1 (b) Display-2

Figure 1.2: Depth image in two different display format



(a) Imaging system of grouped objects (b) Imaging system of distinct objects

Figure 1.3: Image acquisition system

Figure 1.3 shows the importance of depth image over optical image. It shows an image acquisition system for capturing the intensity images and the depth images in two different scenarios i.e. with grouped objects, and with distinct objects. The image acquisition system shown in the figure has two light illuminator (optical and IR illuminator) on the right division and two sensors (RGB sensor and IR sensor) on the left division of the camera rig. The RGB sensor captures optic rays of optical illuminator being reflected off the object surfaces to form an intensity image, and the IR sensor captures the IR rays reflected off the object surfaces to form the depth image. Figure 1.3(a) shows the imaging mechanism of grouped objects where the objects are placed in a group such that their projections on the image plane overlaps to some extent. The intensity image might give a hint of the foreground and the background object based on the hidden portion of the objects, and the depth image also gives a sense of object location from the camera position. Here, the importance of depth image over intensity image is not so clear, because in both the images we could predict the foreground and background object.

However, in Figure 1.3(b), the objects are placed distinct, and under the assumption that the objects are of same colour, then the intensity images could not predict the location of the object, but on the other side, the depth image make more sense of the objects location. Hence, depth image give the location perspective of the objects in the scene.

The high-end depth cameras scan the scene column wise, therefore it is more accurate, but it is time consuming because of its depth estimation approach. Such cameras are not suitable for real-time applications. On the other side, the modern depth cameras scan the whole scene at once, and it is much faster. Such cameras are suitable for use in real-time applications, and they are available at affordable price. But the problem is, these cameras have lower spatial resolution and the images captured are corrupted with noise and sometimes missing regions.

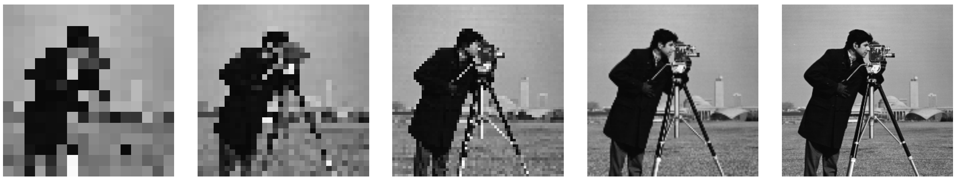
The resolution of the image is directly proportional to the size of the sensor plate. The bigger the sensor plate, the higher will be the image resolution. In 1990's, the cameras were of very low resolution, say 0.3 megapixels (MP), which is a VGA resolution with image size of 480×640 (height $\times$ width). Now a days, the optical cameras are available with higher spatial resolution (nearly 25 MP or more), which is nearly 100 times more as that of depth camera. The huge capacity of sensor plate is because of the development of the VLSI technology which led to the miniaturization of sensors. However, this is not the case with modern depth cameras. The images captured from modern depth cameras generally suffer from low spatial resolution, and most of the time corrupted with noise because of the external conditions, and fewer times it will suffer from missing regions because of occlusion.

For applications like robot navigation, autonomous driving vehicle, etc., low-resolution images will lead to poor performance. Hence, such images need to have increased spatial resolution such that the objects in the image seem distinct with sharp image boundaries. It also needs to be free from noise and the missing regions by some denoising and inpainting methods. These problems are mostly related to modern depth cameras.

Alternatively, one can use high-end depth cameras which can provide high-resolution depth images with sharp edges, but it suffers from low frames per second (nearly ~ 10 fps), which makes it unsuitable for real-time applications. One can still use modern depth cameras which provide low spatial resolution and high fps (nearly ~ 30 fps), and then later use methods to construct higher spatial resolution depth images. The process of increasing the resolution of an image is called super-resolution (SR).

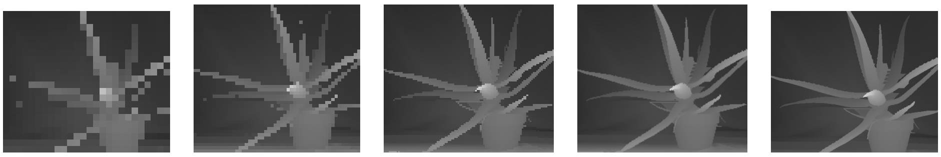
The low spatial resolution problem seems to have two alternate solutions. Either manipulate the hardware configuration to accommodate more pixels (called hardware solution), or perform a software based processing to increase the number of pixels (called software solution). In literature, the software solutions are called the SR method, which takes LR image as input and produce an HR image as output. In hardware configuration, one can either increase the sensor plate size, or decrease the spacing among pixels by placing them close enough to accommodate more pixels per area. The former choice unnecessarily increases the cost of the device and make it more bulkier, and the latter choice introduce the shot noise in the image. Hence, hardware manipulation becomes inconvenient. The alternate solution, i.e. software solution, can be used which is based on processing the captured LR image to estimate an HR image, and it seems to be a good choice.

Figure 1.4 and Figure 1.5 shows increasing spatial resolution of two separate cases. Figure 1.4 shows SR of with optical image (*cameraman* image) as one case, and Figure 1.5 shows SR of depth images (*aloe* image) as another case. In these figures, the image on the extreme left is an image with lowest resolution ($1/16^{th}$ of full resolution), and the image on the extreme right is an image with full resolution. One can notice that, as the spatial resolution increases, the details in the image are also seen more plausibly.



(a) 1/16 (b) 1/8 (c) 1/4 (d) 1/2 (e) Full (256x256)

Figure 1.4: Increasing spatial resolution of intensity image



(a) 1/16 (b) 1/8 (c) 1/4 (d) 1/2 (e) Full (352x416)

Figure 1.5: Increasing spatial resolution of depth image

Super-Resolution (SR)

SR is a process of obtaining a high-resolution (HR) image either from single low-resolution (LR) or from multiple LR images. HR image will always have more number of pixels as compared to the LR image, which makes the HR image more plausible to eye and easy to identify or locate objects in it. For instance, given an LR image $\boldsymbol{Y}$ of size $m \times n$, say, the SR method produces a super-resolved image $\hat{\boldsymbol{X}}$ of size $mq \times nq$, where q is the upsampling factor by which the image was enlarged in both x- and y-directions. Figure 1.6 illustrate super-resolution in the form of pixels on a frame grid, where the first image grid is viewed as LR image, second image grid is an SR image upsampled by 2 ($q = 2$), and third image grid is upsampled by 3 ($q = 3$).[1] Figure 1.7 shows the idea of super-resolution of a depth image, where the input is an LR image of size $m \times n$, and the output is an HR image of size $mq \times nq$.

---

[1] Each box represent a pixel, and they are all of same size in all three cases, which means, the spacing between pixels are equal.

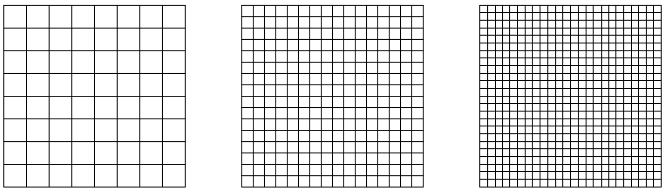


Figure 1.6: Camera sensors of equal size with different number of CCD elements. Resolution of sensors from L to R: 8×8, 16×16, 24×24

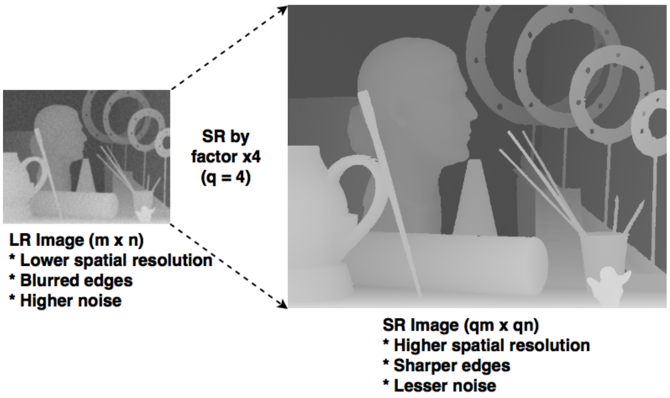


Figure 1.7: Idea of super-resolution with input as LR image and output as HR image.

SR is considered as an ill-posed inverse problem because there does not exist a unique solution for a given LR input. There can be multiple HR outputs for a given LR input. The SR problem is also be viewed as under-determined system where there are fewer equations (pixels in LR image) than the number of unknowns (pixels in HR image), which results into infinitely many solutions. To find the optimal solution, some regularization are enforced into the solution to stabilize the inversion of the ill-posed problem, e.g. smoothness constraints, edge constraints, gradient constraints etc. Figure 1.8 shows the forward process (HR to LR) and reverse process (LR to HR) to illustrate how LR image is formed first, and then how it is used to reverse the operation and obtain the HR image. The forward process is the imaging model which is used in generating LR image from HR image, and the reverse process depicts the SR problem of obtaining the HR image from the LR input image by inverting the model to estimate the factors that produced the input. Eq. 1.2 shows the mathematical representation of SR as an ill-posed problem,

$$y = \mathcal{A}x \tag{1.2}$$

where, y is the observed vector, x is approximate solution which is inaccessible directly, and $\mathcal{A}$ is an operator. The solution to Eq. 1.2 is typically found using $\hat{x} = \mathcal{A}^{-1}y$, where $\hat{x}$ is the exact solution, but since $\mathcal{A}^{-1}$ is not continuous, so there does not exist any solution, or there exist a solution but not stable because a small change in the observed input y lead to large change in output x.

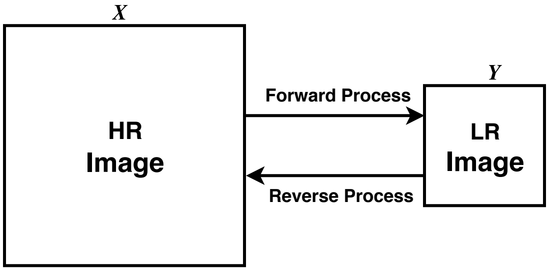


Figure 1.8: Forward and inverse process for HR-LR image

1.2 SUPER-RESOLUTION PREVIEW

Super-resolution (SR) is a class of method for increasing the resolution of an image such that the details in the image are seen clearly. Generally, the images captured from low resolution camera have poor quality as it is difficult to distinguish the objects in the image. The reason is that, the low resolution camera sensor receives many reflected signals from the scene but many gets averaged on the single pixel on the sensor to estimate a pixel value. This happens at all the pixels location on the sensor. Had there been more pixels on the sensor receiving the same set of reflected signals, it would have generated more pixel values for those reflected signals which could have made the object look clearer and distinct.

Resolution plays a crucial role in many scenarios, e.g. in surveillance to detect the persons face, or in medical surgery to locate the operating locations with high accuracy, and many more to list. The recent technology has shown the ability to produce the powerful digital optical cameras with high spatial resolution, but on the other side the depth cameras are not able to cope up with the trend.

Initially, super-resolution techniques were proposed for intensity images where they use multiple LR images to construct a single HR image. Such methods, in literature, are called multiple image super-resolution (MISR). These LR images were assumed to be sub-pixel shifted from each other. Under this condition, each LR images contain some extra information about the scene. Fusing these images helps in extracting the unique information from each input images which helps in reconstructing the HR output image. The quality of the solution found by such techniques could be improved further if there are more number of such LR images, thereby combining unique information from all of those multiple LR images and produces a better HR image.

There are some classical interpolation methods which are used for increasing the image resolution. These are ,nearest neighbor (NN) interpolation, bilinear intepolation, bicubic interpolation, spline interpolation (or bicubic spline), polynomial interpolation, lanczos interpolation, etc. These methods are briefly explained in Appendix A for the sake of brevity.

Other than these classical interpolation techniques, there are several other methods to super-resolve the intensity images, e.g. using sparse representation (Yang et al., 2010; Yin et al., 2013), using gradient prior (Sun et al., 2008, 2011), using training examples (Freeman et al., 2002; Kim and Kwon, 2008), using self training examples (Glasner et al., 2009), using implicit transformed self examples (Huang et al., 2015), using deep learning (Dong et al., 2016; Lim et al., 2017; Kim et al., 2016), etc.

Recently, depth images are gaining popularity because of its need in several applications. As modern depth cameras produce LR depth image and does not cater to the need of the mentioned applications, the existing optical image based super-resolution methods had been tried and tested on depth images to generate a spatially high-resolution depth image. Direct porting of optical based SR methods implemented for optical images are not trivial, because depth images have different properties as compared to optical images. The SR methods on depth images were experimented with some modifications, as the set of methods and approaches proposed for intensity images might not work perfectly for depth images as these images are of different modalities altogether. Many methods have been proposed for depth image SR which preserves the edge discontinuities, because edges are the prominent features of a depth image.

1.3 IMAGE MODELING

Low-Resolution Image Model

The low-resolution image model is a mathematical model which is used to model the low-resolution image from the ground truth (GT) image. This model is used only LR image generations which is used as inputs to the proposed SR methods. This kind of model replicates real-time depth camera environment. There are depth image datasets e.g. Middlebury dataset (Scharstein and Szeliski, 2002)) and few other datasets which contains high-resolution depth images with its corresponding registered colour images of the same scene. The Middlebury dataset provide images with different resolution levels, i.e. from full-resolution images (with approximate size of 1100 × 1300) to one-third resolution (with approximate size of 350 × 450). These images are treated as high-resolution GT images. For obtaining the observed LR images, we apply the LR imaging model on these GT images which is mathematically represented as shown by Eq. 1.3,

$$Y = \mathcal{D}\mathcal{B}X + \mathcal{N}, \tag{1.3}$$

where, X is the HR GT depth image and Y is its downsampled LR depth image which is obtained by applying blurring operation ($\mathcal{B}$) and downsampling operation ($\mathcal{D}$), and added noise ($\mathcal{N}$). The operation matrices $\mathcal{B}$ and $\mathcal{D}$ can be used interchangeably because of their cumulative properties.

Since we are observing the LR image Y as degraded image of the GT image X, the SR method should produce a solution $\hat{X}$, which is as close to the GT image X as possible. This is not the case with real depth camera, as it will have build-in modeling parameters which generates an LR image with all the real effect of low-resolution, noise corruption and missing regions. Since it has no GT image to compare to, so the subjective evaluation is performed for the proposed method for such real-time images.

The Eq. 1.3 used in the thesis is only to synthesize the LR image, and it has not been used in any form in the proposed methods for depth reconstruction or for SR of depth images.

Sparse Low-Resolution Image Model

For dense depth reconstruction and its super-resolution problems, the input has few random visible pixels on the input image. For dense depth reconstruction problem (also called DR), the input and the output resolution is same. In a similar way, super-resolution problem from sparse LR input (also called DRSR), the input is downsampled version of the output resolution by the amount equal to upsampling factor.

For generating input image for DR problem, Eq. 1.5 is used, where y is the GT image and $\tilde{y}$ is the sparse depth input, and $\mathcal{S}$ the sparse generating operator. Similarly, for DRSR problem, Eq. 1.4 and Eq. 1.5 is used to generate the sparse LR image, where the first equation (Eq. 1.5) is used to generate the LR image, and the second equation (Eq. 1.5) is used to generate the sparse LR image.

$$y = \mathcal{D}\mathcal{B}x + \eta \tag{1.4}$$

$$\tilde{y} = \mathcal{S}y \tag{1.5}$$

where, x $(q^2mn \times 1)$, y $(mn \times 1)$ and $\tilde{y}$ $(mn \times 1)$ are lexicographical ordered HR, LR and LR point cloud images respectively, q is the upsampling factor, B $(q^2mn \times q^2mn)$ is blur matrix, D $(q^2mn \times q^2mn)$ is downsample matrix, η $(mn \times 1)$ is a additive noise vector, and S $(mn \times mn)$ is the point cloud sampling matrix with ones and zeros.

For a given sparse LR image $\tilde{y}$ $(mn \times 1)$, we execute the DR module to produce $\hat{\tilde{y}}$ $(mn \times 1)$. The dense depth map output $\hat{\tilde{y}}$ $(mn \times 1)$ is then mapped onto the HR grid to produce $\hat{x}_{mid}$ $(q^2mn \times 1)$, on which the SR module is executed to produce $\hat{x}$ $(q^2mn \times 1)$, where q is the upsampling factor.

1.4 MOTIVATION AND CHALLENGES

Motivation

In past, the obvious techniques to estimate the disparity maps were from stereo images. The stereo images are a pair of left and right viewed images from a slightly different locations which makes these images non-coplanar. For disparity map estimation, first these two images need to be registered such that all the regions in the left image have their corresponding similar region in the right image in a space confined to the horizontal search (Lucas et al., 1981). This approach is very time consuming, and might results in a miss correspondence at smooth regions which give rise to errors in final disparity map. There were many modifications being done to avoid the miss match by considering the bigger patches to render accurate match correspondence by utilizing lesser search space. All these disparity estimations methods are pixel based methods as disparity values were computed by minimal aggregating value at each pixel. There are other reasons like occlusion, large saturated areas and repetitive patterns, thats why stereo correspondence still lack in generating accurate 3D (one can arguably call it 2.5D). There are recent methods on deep learning (Luo et al., 2016) which finds the marginal distributions over all possible disparities for each pixel, and it is viewed as multi-class classification. Since these disparity map from stereo images will be of same size as that of the size of the stereo image, the spatial resolution of the disparity map is not a worry, but still it will be corrupted with irregularities at the edges.

Since disparity map estimation from stereo images is a challenge, the recent trend to capture 3D information of a scene is by using ToF depth cameras because of its convenient use and low cost. However, these modern depth cameras capture depth images which suffer from low spatial resolution, noise and some issues of missing regions, as discussed earlier. Moreover, depth images are used in many applications including some real-time applications, e.g. in applications related to robot navigation and autonomous driving vehicle, depth images are necessary to guide the system for obstacle detection, localization and decision making. In machine vision, the depth images are used for various purposes, e.g in surveillance to determine the present/absence of person, or count the number of persons/objects in the zone. In medical domain, the patient positioning and movement monitoring for therapy is observed using depth images. Other applications which include measuring and detecting goods, or for human safety to monitor workers in proximity of dangerous equipments, all need depth images to have make accurate and effective decisions. The depth image based applications demand high-resolution depth images to have better accuracy which neither the modern depth cameras meet the requisite criterion to serve the purpose nor the existing classical interpolation techniques provide better results because of its implicit averaging It ultimately led to the class of depth image super-resolution to fulfill the need-of-the-hour by providing super-resolved depth images with improved edge discontinuities.

Challenges

The main challenges in any SR method is to retain the image details. The image details are mainly composed of edges, corners, and textures. Most of the existing methods try to retain the image details as maximum as possible, but it becomes more challenging with increasing upsampling factor.

Precisely, depth images lack texture features, but most importantly, the prominent features in the depth images are the depth edge discontinuities and depth precision. Depth edge discontinuities are seen at the edges of the objects where they are placed at different depth levels. Depth precision is seen on the side walls, lower floor and upper ceiling where the depth linearly increases as we move deep into the scene.

As discussed earlier, depth images not only suffer from lower spatial resolution, but they also suffer from noise and missing regions. The noise is introduced majorly because of the external parameters when IR signal is reflected back from the object surface to the IR sensor. There can also be internal parameters which are internal to the cameras mechanical system, which when in operations introduces noise. As the exact distribution of noise is unknown, denoising the depth image is a challenge. However, there are sufficient work which have considered random noise distribution and assume that the original image was corrupted with similar kind of noise. Another issue is the the missing regions issue, which can either be an irregular missing region or a regular missing region. Irregular missing regions are because of unreachable locations or reflecting surfaces in the scene, and regular missing regions are at the object boundaries because of the scene occlusion.

1.5 RESEARCH OBJECTIVES

Based on the literature review and the challenges and short coming of existing SR methods, the following research objectives were framed to push the results to a better level. The research objectives composed for this thesis are related to depth image restoration and enhancement.

Objective-1: The first objective is to propose a wavelet transform based single depth image SR method. Wavelets are well known in the field of image processing for denoising problems by extracting features from a finer level to a coarser level. With this hope, the first objective is to extract some edge features in multiple directions and use them to obtain an SR image.

Objective-2: Another objective set out is to propose an SR method for depth image by utilizing the corresponding HR colour guidance image segment cues. As most of the existing intensity SR methods work with an initial bicubic interpolated image or the LR points spread sparsely in uniform way on HR grid, the objective is to follow the same practice of bicubic interpolated images or sparsely spread LR points on HR grid as starting image, but by utilizing the segment cues from the corresponding HR colour guidance image.

Objective-3: Depth reconstruction from sparse depth data is a challenge, and this scenarios is very much useful for low transmission bandwidth setting where there is not enough channel space to send the complete data. Instead, only a sparse (very few) depth data is send. This scenario becomes more challenging at the receiving side as it needs to reconstruct the full image with only sparse depth data available. Further, the aspiration is to use the same concept of guidance colour image for super-resolving an LR image with only few random sparse points. The inherent objective of this method is also to see if it can be applied to address other depth image related problems like depth image denoising and depth image inpainting.

Objective-4: Learning HR-LR relationship helps in knowing the implicit linking of the LR patch with its associated HR patch. The next objective is to learn such HR-LR relationship from the set of training images., and use the learned HR-LR relationship to compute the SR image of an unseen LR image Such techniques have been heavily used in the field of SR, and rendered promising results.

Objective-5: The iterative SR methods performance majorly depends on its input. As many such methods either take bicubic interpolated image or the sparse points uniformly spread over HR grid. The aim is to use some better initial input which is faster and easy as bicubic interpolation.

1.6 IMAGE DATASET

There are few freely available dataset of depth images which is widely used in the field of depth image super-resolution and the one which is well suited for our work is the Middlebury depth image dataset (Scharstein and Szeliski, 2002, 2003). This dataset has several varieties of collection of depth images varying from depth images of planar object to the images of objects of varied shapes and sizes. As discussed earlier, it has images of three different resolutions, i.e. full resolution, one-half resolution, and one-third resolution. Another dataset which has both depth images and its corresponding colour images is *Kitti* dataset. The images in this dataset are synthetic images of outdoor scenes.

There are some synthetic depth image, refer Mac Aodha et al. (2012), which we have explicitly used for training purpose. Because, these images are of bigger resolution ($\sim 800\times800$) with sharp edges at the object boundaries. There are also few real-time depth images in low resolution available for free download, which was proposed by

Ferstl et al. (2013) in their work. These images are available with their corresponding HR colour images.

Apart from freely available depth images from dataset, we have collected few real-time depth images from Kinect depth camera solely for the purpose of testing the proposed SR methods.

1.7 PERFORMANCE METRICS

For comparing the SR results with the GT image, we chose peak signal to noise ratio (PSNR) and structural similarity (SSIM) (Wang et al., 2004) as the performance metrics to show the effectiveness of the proposed method in generating the SR output image. Mean squared error (MSE) performance metric is also used to measure the performance of the proposed SR method. Metrics MSE and PSNR are calculated between estimated output image and the ground truth image. These are pixel based goodness measure of the estimated output and the ground truth. The mathematical formulation of MSE and PSNR is represented in Eq. 1.7 and Eq. 1.6 as,

$$PSNR = 10\ log_{10}\left(\frac{MAX_I^2}{MSE}\right) \tag{1.6}$$

$$MSE = \frac{\sum_{i,j}(\hat{I}(i,j) - I(i,j))^2}{m \times n} \tag{1.7}$$

where, $\hat{I}$ is the estimated output and I is the ground truth image of size $m \times n$, MAX_I is 255 for an 8-bit image, and $l(\cdot)$, $c(\cdot)$ and $s(\cdot)$ are the luminance, contrast and structure functions respectively.

SSIM is another performance metric which is used in our work to measure the performance of the proposed SR methods. Unlike pixel based measure, SSIM considers the luminance, contrast and structure between the estimated output and the ground truth image. SSIM measure is more close to the the subjective measure as it incorporates the sense of human visual system (HVS). The mathematical equation if represented as shown in Eq. 1.8.

$$SSIM = f(l(x, y), c(x, y), s(x, y)) \tag{1.8}$$

The qualitative measure used here are full reference image quality assessment (FR-IQA) measures, as it require the reference image to quantify the quality of the output image.

1.8 THESIS CONTRIBUTION

The contribution of the thesis is as follows:

- Wavelet transforms (e.g. DWT and SWT) are explored along with the combination of gradient information in horizontal and vertical direction to extract image details from an input LR image for producing better and sharp super-resolved output HR image.
- For SR related problems, the HR guidance colour image corresponding to the same scene for LR depth image has shown good results in super-resolution domain in the recent past. So the use of HR guidance colour image has been explored to provide local segment cues obtained by applying simple and robust segmentation methods like mean-shift (MS) algorithm or simple linear iterative clustering (SLIC) algorithm on the guidance image.
- A similar approach of using HR guidance colour image has been explored for the scenarios where the extra step of computing an initial HR estimate is not considered. In such case, the input LR image will be treated as sparse LR image when laid on the HR grid.
- The use of colour guidance image which can provide sufficient cue has been explored for dense depth reconstruction from very sparse depth data. The guidance image is used only to extract the local information in the image based on the colour information. However it is utilized under the assumption that the objects at different depths have different colours.
- There have been many approaches on super-resolution from input LR image, but SR from sparse LR has seen less growth mainly because of its challenging task. It is very much useful if it is required to send lesser data (sparse LR image) and still reconstruct the HR image at the receiving end. Towards this objective, the use HR guidance colour image has been explored for dense reconstruction and its super-resolution in a single framework.
- Learning LR-HR relationship from the training exemplar images have been explored using Gaussian mixture models (GMM). The learned model is then used to perform the super-resolution task on a given input LR image.

- Other common problems of modern depth cameras i.e. denoising and inpainting, have also been addressed using the guidance colour image. For denoising, different levels of noise have been considered, starting from low noise standard deviation of 1 to high noise standard deviation of 10. For depth image inpainting problem, different types of missing regions have been taken into consideration, e.g. random missing region, structural missing region, real time Kinect captured depth images and random hand scribbled missing regions.
- Better initial tentative estimates are important for iterative methods. For depth image SR problem, a cascade approach has been explored, which is constructed by combining residual interpolation (RI) method and anisotropic total generalized variation (ATGV) method in a single framework. Here, the RI method output is used as an initial estimated HR output, and the ATGV method produces an accurate HR output in an iterative manner over the previously generated HR output.
- The proposed methods have been experimented on depth image from multiple dataset without noise (σ=0) and with added noise (σ=5). The SR results are shown for SR upsampling factor $\times 2$, $\times 4$ and $\times 8$ in almost all experiments.

1.9 THESIS ORGANIZATION

The thesis is organized as follows:

Chapter 2 presents some of the existing depth image super-resolution and depth reconstruction methods. This chapter briefly review the existing literature, and categorize the depth super-resolution and depth reconstruction methods into different classes.

Chapter 3 presents a wavelet based single depth image super-resolution. It uses discrete wavelet transform (DWT), stationary wavelet transform (SWT), and the gradient information of the initially interpolated LR image. Using this information collectively, an intermediate stage has been proposed to enhance the high-frequency subbands to recover the HR image for both noiseless and noisy scenarios. The proposed method has been validated on Middlebury dataset for different upsampling factors (i.e. 2, 4 and 8), and it is shown to be superior when compared to some related DWT and SWT based SR methods. Encouraging performance of the approach has also been demonstrated on noisy depth images also.

Chapter 4 demonstrates the use of HR guidance colour image for depth image super-resolution. The presented method can be categorized into two classes based on the type of the input it takes. First type of input is an LR depth image which is bicubically interpolated, and the second type of input is an LR depth image mapped onto the HR grid of desired resolution with equal spacing between the known depth pixels. In this work, the HR colour image is segmented using well-known segmentation approaches such as MS or SLIC segmentation approach. The approach begins with a highly over-segmented color image. Using these local segments as cue to estimate the depth values, a median filling approach is employed on initially estimated SR image. The presented method also demonstrate hierarchical approach for higher upsampling factors. A bilateral filtering is followed as an end module in the SR pipeline to remove any artifacts at the abutting regions of the local segments by carefully preserving the edge discontinuities.

In Chapter 5, a relatively simple and efficient methods for depth reconstruction from very sparsely sampled random depth data has been presented. The proposed methods exploit the segmentation cue obtained from a registered colour image of the same scene. The depth reconstruction method uses two different approaches to estimate the unknown pixels. First approach is the plane fitting (PFit) approach, which involves cost computations on plane-fitted on depth values over local segments; and second approach is the median filling (MFill) approach, which computes median of depth values in a local segment region. It utilizes MS and SLIC segmentation methods for segmenting the guidance colour image. Results of dense depth reconstruction from as low as 1% of available depth data has been presented. The variant methods presented here has been compared with recent related state-of-the-art method and shown results both qualitatively and quantitatively. This chapter also presents the the problem of super-resolution from sparse LR input (DRSR). It combines DR and SR problem together. It poses comparatively more challenge when compared with DR or SR problems alone. For SR with higher upsampling factors, a hierarchical approach has also been presented. It is also shown that the proposed guidance based depth restoration method can also be used to address other depth image related problems like depth image denoising and depth image inpainting. For denoising problem, various levels of noise has been considered and shown the comparable results with other standard denoising methods. For inpainting problem, various kinds of missing regions have been considered like random missing region, structural missing region, real time Kinect captured depth images and random hand scribbled missing regions, which demonstrates the robustness of the methods.

In Chapter 6, a Gaussian Mixture Model (GMM) based single depth image super-resolution method is presented. GMM has proven to be a good model for unsupervised clustering method which is based on the probability distribution. It has been widely used in image restoration, clustering and regression problems among others. The advantage of this unsupervised GMM technique has been utilized to address the problem of single depth image SR. For training the GMM model, a set of HR and LR image pairs are concatenated to form a matrix which is fed to GMM model for training. The inherent relation between the HR and the LR patches are captured by the covariance matrix which helps in deriving the HR patch for the input LR test patch. Further, Expectation-Maximization (EM) iterative algorithm is adopted to estimate the parameters, which guarantee the convergence of the Gaussian modeling on the given data of pair of HR and LR patches. For higher upsampling factors, a hierarchical approach of GMM training has been demonstrated. The learned GMM model is tested on several depth images from Middlebury dataset. The effect of GMM training over different number of Gaussian mixtures has also been studied. The upsampling results are quantified based on qualitative and quantitative performance metrics.

Chapter 7 describes the proposed initial estimate for iterative SR methods. The initial estimated SR output plays a crucial role in obtaining the finer SR output with rich details. Since the iterative SR methods use the output from their previous step and use it in their next step, the output in the previous step should have maximum details which can be transferred to the next step. The proposed method used residual interpolation (RI) method as an initial stage which gives a better initial SR output as compared to other methods. The output of RI method is then fed to the next stage. The next stage is a anisotropic total generalized variation (ATGV) method which is proven to be good for iterative SR methods. It utilizes the anisotropic diffusion tensor to guide the upsampling. The proposed cascade approach of RI with ATGV gives better results as compared to these methods when considered alone.

Finally, Chapter 8 concludes the thesis and some future directions.

All the proposed method for either depth super-resolution or for depth restoration uses different techniques to achieve the task. The first method proposed was SRDWT which utilizes the implicit information from a single input LR depth image. It estimates the high-frequency details to get the super-resolved output HR depth image. When the super-resolution factor requirement is small (say $\times 2$) and there is insufficient computational resources, in those cases the wavelet based SR method would be the best choice. It might not achieve good results for higher upsampling factor because it provide very less information as we decompose the image to higher levels (higher than 1).

If the upsampling factor is large, one has to look for some extra input of source to guide the super-resolution process. In such cases, the proposed HR guided image based super-resolution (LRBicSR or LRSR) can be used. The only requirement of this method is it needs two input, one is the LR depth image itself, and other is the HR guidance colour image of the same scene.

There can be some instances where capturing the dense depth map need more time because of the cameras high computation time. Instead, one can capture only sparse depth map and restore the dense depth map. Here, the guided image of the same scene will provide the cue for dense depth restoration.

Instead of using guidance image for cue, there are some recent methods which makes use of training images. If there are enough training images to train a model and enough computation resources to perform training, then one can choose the proposed SRGMM method to get better SR results. GMM will be first trained with known images of LR-HR patch pair, and in the testing phase the learned model try to predict the HR patch for the corresponding unseen LR patch.

In few of the proposed methods discussed above, the bicubic interpolation was used to get an initial SR estimate. If there is some better initial estimate which gives improved results and as computationally less expensive as bicubic interpolation, it would be the best choice for iterative methods for SR. The proposed method ATGVMod does exactly the same. The use of residual interpolation (RI) gives an initial estimate of SR, which is then fed to ATGV module to iterative SR. Such a cascade network converge faster.

CHAPTER 2

LITERATURE REVIEW

[1] Super-resolution (SR) is a well studied topic in image processing. SR is an interesting topic and its inception in the field of image processing and computer vision is around three decades ago, which was first exercised on intensity images. SR operation was performed by collecting multiple LR images which are sub-pixel shifted and super-resolving the reference image, and it is well explored. However, super-resolution for depth images has been a relatively recent exploration. With the recent demand for depth images and its related application, the focus has now been shifted to depth image SR.

There are various approaches used to perform depth image SR task. These methods can be broadly classified into different classes as single depth image SR methods, multiple depth image SR methods, guidance based depth image SR methods, training based depth image SR methods. The existing methods in each of these SR classes have been explained briefly.

Other than SR methods, there are several other existing depth restoration methods for handling other depth image related problems like sparse depth image, noisy depth image, and depth image with missing regions. These depth restoration methods can be broadly classified into dense depth reconstruction from uniform samples, and non-uniform samples, depth denoising methods, and depth inpainting methods.

2.1 SINGLE DEPTH IMAGE SR METHODS

As mentioned earlier, the interest of depth image super-resolution came with its recent demand in depth image related applications. At the beginning of this growing research

---

[1] Chandra Shaker Balure, and M. Ramesh Kini. ”A Survey–Super Resolution Techniques for Multiple, Single, and Stereo Images.” *Fifth International Symposium on Electronic System Design (ISED-2014).* IEEE, 2014.

field, the existing methods of single intensity image super-resolution were adopted for single depth image super-resolution, but the usage was not so trivial as both the intensity and depth image have different properties.

From the intensity image super-resolution perspective, single image super-resolution method would only require a single image as an input. The SR method would try to super-resolve it by looking at the intrinsic information in the image. A similar approach has been followed by Demirel and Anbarjafari (2011a) who propose single intensity image super-resolution. It was mainly focused on super-resolving the satellite images. They proposed an intermediate stage to estimate high-frequency content by adding the difference between the interpolated low-frequency subband of the DWT operation on LR input image and the LR input image itself to all the interpolated high-frequency subbands.

A similar work on super-resolution for intensity images was proposed by Demirel and Anbarjafari (2011b). This method adds the high-frequency components of the SWT of LR input image to the interpolated high-frequency component of DWT on LR input image to recover the high-frequency details.

These methods mentioned from literature were proposed for intensity images. As we have used these methods to compare it with our proposed single depth image super-resolution methods, we have discussed these here.

In literature, there are many research papers which claim to be a single depth image super-resolution methods, but intrinsically they use some kind of learned parameters from the some other datasets. For the reason that this section presents only those method which purely uses single image as input without any additional information, those methods which claim to be single image super-resolution methods have been discussed in other sections below.

Performing super-resolution with only single image is not so trivial, so the work in this class is very limited. The conventional approach to increase the resolution of the image (without any additional input or information) is the interpolation method. There are various kinds of interpolations methods, e.g. nearest neighbor interpolation, bilinear interpolation, bicubic interpolation, lanczos interpolation and more. These methods server the purpose of increasing the resolution, but the resultant image suffer from blurring artifacts as these methods smoothen the image details. They act like a low pass filter, where the high frequency contents in the image, like sharp edges and corners, get smoothened, but it reduce the noise (as noise is high frequency content) to some extent.

2.2 MULTIPLE IMAGE DEPTH SR METHODS

Super-resolution from multiple images require multiple LR images. One of the LR image is considered as a reference image which needs to be super-resolved by fusing unique information from other LR images. This method can be advantageous if there are multiple such LR images, and they are sub-pixel shifted with each other and the reference image. These methods are also considered as motion-based SR methods because the LR images which are used as inputs are the result of the motion (slight motion) of the camera or the hand while capturing the images. The non-redundant sampling information from the other LR image is fused together to estimate the information at the unknown pixel on the HR grid. The assumption is that each LR image would contain some unique information due to their subpixel shifts.

The work of Schuon et al. (2008) show the increase in the X-Y measurement resolution by utilizing several depth images captured from a minimally displaced viewpoints. These images were then aligned to bring them to the same plane and subsequently combining them to produce a high-resolution depth image.

$$\hat{\mathbf{X}} = \underset{\mathbf{X}}{\operatorname{argmin}} \left[\sum_{k-1}^{N} \parallel D_k H_k F_k \mathbf{X} - \mathbf{Y_k} \parallel_p^p + \lambda \Upsilon(\mathbf{X}) \right] \tag{2.1}$$

where, $\mathbf{X}$ is the original depth image, $\mathbf{Y_k}$ are the LR depth images for $k = 1, \cdots, N$, D_K is the decimation matrix for downsampling, H_k is blur matrix to introduce blur, and F_k is the translation operator, $\Upsilon(\mathbf{X})$ is the regularization term with its weight λ. This minimization problem provides the solution as super-resolved depth image of the scene.

The work of Gevrekci and Pakin (2011) is also to increase the spatial resolution of the depth (or range) image of ToF camera. They use novel multi-exposure data acquisition technique with different integration times. The image with less integration time gives better foreground and noisy background image. However, image captured with more integration time capture reliable background but with the expense of saturation in foreground. They use projection onto convex sets (POCS) algorithm for image reconstruction. Their proposed method apply heuristically selected amplitude constraint sets on depth image which can be further improved by calibration based constraint set formation.

A work of Bhavsar and Rajagopalan (2012) proposes two method to address the problems related to lower spatial resolution of the captured image from the low-cost scanner, and the problem of long acquisition time of high quality scanners. Their first method uses multiple LR range images which are relatively-shifted. The motion between these LR images is served as cue to super-resolve the image. The proposed SR framework models HR images as Markov random field (MRF). The solution is constrained by using inhomogeneous MRF priors. Their method, however, require multiple LR images and their accurate motion w.r.t. the reference image. Their second method facilitate the densely depth reconstruction from sparsely measured range data to combat the long acquisition time of the high-quality scanners.

Work of Kil et al. (2006) proposed a method to improve the surface resolution by capturing multiple scans from laser range scanner, and then later combining them to produce a super-resolved image. The subsequent scans were randomly shifted so that each scan contributes slightly different information to the final model. As they fuse information from multiple scans, they claim that the noise gets reduced . As mentioned earlier, such methods require large number of LR scans, which is a challenge. Also, these methods could not help much to achieve higher SR upsampling factors which limit its use for applications requiring higher factors of super-resolution.

2.3 GUIDANCE BASED DEPTH IMAGE SR METHODS

Guidance based depth image super-resolution (DISR) methods are the methods which make use of a secondary images as a guidance image to super-resolve the LR depth image. Generally, the guidance image would be the input image itself which is used to estimate the weights of the filters (i.e. domain filter and range filter) as proposed in bilateral filtering (BF) (Tomasi and Manduchi, 1998). This edge preserving smoothing BF filter estimate the *domain* and *range* kernel based on geometric closeness and photometric similarity respectively. The domain kernel refers to closeness of pixel values, and range kernel refers to similarity of pixel values. However, it estimate these kernels from the same input. Few years later, guided image filtering (GIF) (He et al., 2010) was proposed to filter the image by using the local linear model. The guidance image can be the input image itself or another different image. Such methods are becoming popular because of the easy availability of a rig with two cameras placed side-by-side, with one as LR depth camera and other as HR optical camera (e.g. rig of three cameras in Li et al. (2008)). These camera rigs are cost effective, and can be directly used for depth image super-resolution. As the viewpoint of both the cameras are different, it results in misalignment of image frames. However, it can be taken care by well established image registration methods using calibration techniques. However, there are camera rigs like Kinect, which itself produces registered images.

The DISR methods which use a corresponding HR RGB image of the scene as a cue have been shown to be quite effective. The work of Diebel and Thrun (2005) super-resolves an LR depth image by integrating the HR color image in Markov random field (MRF) graphical model. The intuition behind MRF is that the discontinuities in depth image often co-occur with the intensity changes in the intensity image. Further, Yang et al. (2007) employs bilateral filtering on the cost volume computed with the help of the HR colour image, and Kopf et al. (2007) proposed a joint bilateral upsampling (JBU) approach where the intensity kernel is applied on the HR colour guidance image, and it is used to integrate the high frequency information from the guidance image into the low resolution depth image. However, JBU method suffers from the problem of texture copying in the region where there is noise in the smooth region. To prevent this, an extension of JBU was proposed by Chan et al. (2008), which is termed as noise-aware filter for depth upsampling (NAFDU). Their filter design is in such a way that it considers the small neighborhood around the pixel and decide on the filter weights by using a blending function. Yadav et al. (2014) proposed an improved local approach using associated color image over GIF to super-resolve the depth image. A method based on global energy minimization (Ferstl et al., 2013) calculates anisotropic diffusion tensor based on the HR color image and they build their method using MRF and least squares optimization by incorporating higher order regularization. Such a global approach, yields good results, but the global regularization based optimization makes it computationally intensive. The geodesic distances which was used in image processing applications has also been used for upsampling the depth image Liu et al. (2013) by finding the affinity measure between the two points (the known points in the sparse HR depth image grid and the unknown depth points) using geodesic distance. Yang et al. (2013) proposed a method which combines median filtering and BF filtering, named joint bilateral weighted median (JBM) filter, for the problem of depth upsampling in an hierarchical fashion, which claims it to be improving the upsampling accuracy and reduces the computational complexity.

Later, Garcia et al. (2010) proposed an extension to the JBU method to get away with the texture copying problem. He found that, limiting the prior information only from the guidance image itself is not sufficient, hence, he used depth image values also in estimating the range kernel. He proposed an addition factor for the filter kernel, called credibility map (CM), which is based on the gradient information of the LR depth input, which assigns lower weights to the pixels along the strips of depth edges. By using their pixel weighted average strategy (PWAS), they fuse the depth data together for depth upsampling. Further, Garcia et al. (2011) have proposed a filter which uses credibility weight of a pixel to decide whether to use the PWAS filter which uses only guidance image or to use the same PWAS filter but with considering only depth information, and Garcia et al. (2015) have proposed a unified multi-lateral (UML) filter where the reliability weight decides whether to consider kernel with intensity image (PWASI) or kernel with depth image (PWASD), and thereby improving the accuracy within smooth regions. Kim et al. (2010) proposed an additional kernel term to the JBU filter which weigh the similarity in the input depth image. Hua et al. (2016) exploit local gradient information of input depth image to deal with texture copying problem of JBU. Park et al. (2014) address depth map upsampling and completion problem by combining the non-local structure regularization with edge weighting scheme. Yang and Wang (2012) combines GIF approach and reconstruction constraints to generate the final HR depth image. Lu and Forsyth (2015) uses HR colour guidance image to extract segments boundaries and corresponding depth boundaries from the co-aligned depth image, and each segment in depth image is reconstructed independently using their smoothing method. Xiao et al. (2015) proposed defocus deblurring and super-resolution of ToF depth image by regularizing the solution in amplitude and depth space directly.

2.4 TRAINING BASED DEPTH SR METHODS

There are another set of SR methods which does not require HR guidance image. Instead, these methods require a set of HR and LR images (called training dataset) to learn their intrinsic relationships. Typically, the training dataset consists of large number of images. Such methods in literature are called learning-based DISR methods. These kinds of method use external training images to learn image details, and then use the learned knowledge to reconstruct the HR image of a new unseen LR input. The training process is typically computationally intensive and the quality of the output depends mainly on the training images used.

For optical image super-resolution problem, training images based SR methods were initially proposed for learning implicit relation between the set of HR and LR images. For instance, the work of Jiji et al. (2004) is based on learning the wavelet coefficients at finer scales between the the HR and the LR images. Sun et al. (2008)

learning some edge priors from HR and LR dataset. Other work in the class of training image based SR methods is by Moon et al. (2015) which uses these training images to cluster the wavelet subband patches using Neighbor Intensity of Local Binary Pattern (NI-LBP). The patches from test images are matched with these clusters using Subtraction of Center from Neighbors - Mean Square Error (SCN-MSE), and the patches from the closest clusters are used to reconstruct the HR image.

A similar approach for learning the HR-LR relationship has also been explored for depth image SR problem. Mac Aodha et al. (2012) proposed an algorithm for increasing the resolution of solitary depth image from synthetic training database. For a given LR patch, it search for the appropriate HR patch from the database. HR patch selection is posed as MRF labeling problem. In another work of Li et al. (2014) using training examples, they disassemble the LR image into parts by matching similar regions from HR training examples, then assemble these corresponding matched counterparts. Xie et al. (2016) propose the combination of guidance and training based depth SR method which is based on patch synthesis. They use patches of edge maps retrieved from HR training images to obtain an HR edge map through a Markov random field optimization, and this HR edge map is used as guidance image for depth SR using modified joint bilateral filter. Recently, deep neural networks have also shown promising results in the area of depth super-resolution by Song et al. (2016), but these methods require huge amount of training data and huge computation resource for training over several days, however, our method does not need either huge training data nor the computation resources.

2.5 DEPTH RESTORATION FROM NON-UNIFORM SAMPLES

Depth image super-resolution problem can be considered as depth reconstruction problem, where the input LR image can be considered as *uniformly* distributed sparse samples on the HR grid. Hence, for the sake of brevity, the super-resolution methods from uniform samples which were discussed in earlier sections are not discussed in this section.

Here, we would consider the methods which were proposed for depth reconstruction from non-uniform samples. Non-uniform samples comes from randomly picking up the depth samples from the scene, and these depth reconstruction method would try to produce a dense depth map. Towards this, following are some methods from literature which does exactly the same. They consider the input image which has non-uniform samples in it.

The work on depth image reconstruction reported by Bhavsar and Rajagopalan (2012) consider examples involving uniform as well as non-uniform sampling. Mandal et al. (2017) perform depth restoration from less sample uses learned dictionary and edge preserving constraints, but this method is computationally expensive unlike the proposed methods which are simple and efficient segment based depth reconstruction. Mandal et al. (2017) follow sophisticated approaches involving sparse representation based methods or constructing sub-dictionaries from exemplar images.

A depth reconstruction by Liu et al. (2015), which is called alternating direction method of multipliers (ADMM), considers sparse representation on wavelet and contourlets, whereas the depth map restoration from under-sampled data (DR-DRU) (Mandal et al., 2017) is based on sparse representation by constructing sub-dictionaries from training images.

CHAPTER 3

WAVELET TRANSFORM BASED SINGLE DEPTH IMAGE SUPER-RESOLUTION

3.1 INTRODUCTION

[1] [2] Wavelet is a brief oscillation which starts from zero, increases and then decreases back to zero with average value of zero. Wavelets are crafted to have some specific properties, which when combined with the portion of the signal, used to extract the information from that unknown signal. Wavelet transforms (WT) are preferred over Fourier transforms (FT) because WT captures both frequency and time information in signal (or frequency and location information in image), whereas FT captures only the frequency information by transforming the view of the signal from time-base to frequency-base. WT can be considered as windowing technique with variable window size with long time intervals for more precise low frequency information, and shorter time intervals when high frequency is needed, as opposed to short-time Fourier transform (STFT) which uses fixed window size. Figure 3.1 show the wavelet transformation with time on one axis and frequency on another axis.

Wavelets were mostly used for compression and denoising, however, wavelets have also been used in the field of super-resolution of optical images. Because of the gaining popularity of depth images and its use in applications like robot navigation, human machine interaction (HMI), automotive driver assistant and many more, it is necessary to provide HR depth images to these application for better outcome. However, the

---

[1] Chandra Shaker Balure, and M. Ramesh Kini. ”Depth Image Super Resolution - A Review and Wavelet Perspective.” *Computer Vision and Image Processing (CVIP).*

[2] Chandra Shaker Balure, M. Ramesh Kini, and Arnav Bhavsar. ”Single Depth Image Super-Resolution via High-Frequency Subbands Enhancement and Bilateral Filtering.” *Eleventh International Conference on Industrial and Information Systems (ICIIS-2016)* IEEE, 2016.

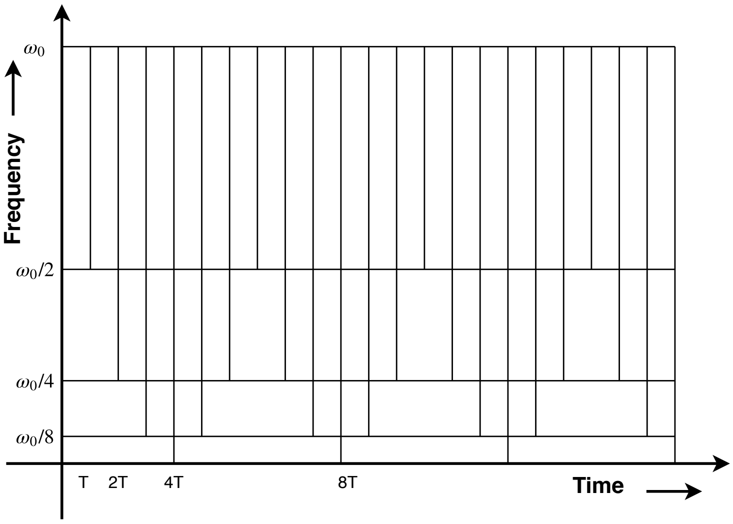


Figure 3.1: Wavelet transformation

modern depth cameras (e.g. Mesa Swiss Ranger, CanestaVision, Kinect, etc.) could not meet the requirement of providing HR depth images, as the depth images captured from these cameras suffer from lower spatial resolution, noise, and missing regions. Hence, there is a need of technique which can produce HR image which is free from noise and missing regions.

Super-resolution methods satisfies the requirement of providing HR images by processing the LR images by retaining the image details, e.g. edge discontinuities. The SR methods for depth image typically target for larger upsampling factors e.g. $\times 4$, $\times 8$ or even $\times 16$ because the depth images have edges as the prominent features, not the texture unlike optical images (Yadav et al., 2014; Ham et al., 2015; Hua et al., 2016).

This chapter address the problem of SR from a single LR depth image. Interpolation methods also use single image to super-resolve it to higher factors, but it fails to preserve fine details in an image as it involves smoothing operation. To overcome smoothing of high-frequency details in an image, many sophisticated methods have been reported in literature to improve the interpolation results. Here, a simple and efficient method for single depth input super-resolution has been proposed as opposed to other existing depth image SR methods which require extra input in addition to the given LR image. The additional input can be either in the form of RGB images, training data, or multiple

LR images.

In literature, there are some wavelet based optical image (not depth image) SR methods and they operate only on a single LR image (Demirel and Anbarjafari, 2011a,b). Similar to Demirel and Anbarjafari (2011a,b), there are some more methods which consider only the LR image as input, which are New Edge Directed Interpolation (NEDI) by Li and Orchard (2001), Wavelet Zero Padding and Cycle Spinning (WZP-CS) by Temizel et al. (2005) and Complex Wavelet Transform Super-Resolution (CWT-SR) by Demirel and Anbarjafari (2010). However, these are relatively traditional as compared to Demirel and Anbarjafari (2011a,b).

The methods of Demirel and Anbarjafari (2011a,b) are closely related to the proposed approach in terms of the number of inputs. However, there are important methodological differences, such as using interpolated images, gradients, and bilateral filtering. The proposed method is presented for depth images, unlike Demirel and Anbarjafari (2011a,b) which involve optical images. The depth image SR importantly involves large upsampling factors (e.g. even up to 8 or 16). It is demonstrated that as the upsampling factor increases, the proposed method shows larger improvement over Demirel and Anbarjafari (2011a,b) as the upsampling factor increases. The performance of proposed method on noisy images has also been shown, which has not been demonstrated in Demirel and Anbarjafari (2011a,b).

The proposed method uses Discrete Wavelet Transform (DWT), Stationary Wavelet Transform (SWT), and the gradient operation on the interpolated LR depth image to recover the HR image. The proposed method is an intermediate stage to enhance the high-frequency subbands to recover the high-frequency information for HR image reconstruction. Iin the later stage of SR pipeline, bilateral filter (Tomasi and Manduchi, 1998) is employed to reduce the residual noise by preserving the edges gained in the previous steps. The proposed method has been validated on Middlebury dataset (Scharstein and Szeliski, 2002) for different upsampling factors (i.e. 2, 4, and 8). It is shown that the proposed wavelet based SR method is superior when compared to DWT and SWT based SR methods (Demirel and Anbarjafari, 2011a,b), which unlike the presented method have considered only noiseless optical (not depth) images.

3.2 PROPOSED METHOD FOR WAVELET BASED DEPTH IMAGE SR

The proposed wavelet based SR method tries to enhance the high-frequency components in the input image. the enhancement is done by extracting multiple implicit information like the high-frequency components in different directions by using DWT (Mallat, 1999), SWT and gradient operations. All these operations are performed on bicubicly interpolated LR image instead of the LR image itself. Use of bicubic interpolation helps in noise reduction, but it blurs the important details in the image. The lost information can be recovered to an extent by the proposed intermediate stage of high-frequency content enhancement. Lastly, the bilateral filter (Tomasi and Manduchi, 1998) is employed at the last stage of the SR pipeline, which suppresses the residual noise and importantly retains the edge information gained from the proposed method. Here, the proposed method is referred to as SRDWT method.

The block diagram of overall proposed method is shown in Figure 3.2. The proposed method is divided into four stages, which is initial estimate, high-frequency combination, image reconstruction and bilateral filtering. Given an LR depth image (D_{LR}) of size $m \times n$, the proposed wavelet based method super-resolves it to a high-resolution image (D_{HR}) of size $\alpha m \times \alpha n$, where α is the SR upsampling factor.

3.2.1 Initial Estimate

Given an LR depth image (D_{LR}), the proposed method employs a bicubic interpolation as a first step as opposed to (Demirel and Anbarjafari, 2011a,b) where they apply DWT and SWT operation on the input LR image itself. Interpolation is the method of estimating the unknown pixels from known pixels. The lower order curve fitting like bilinear or nearest neighbor interpolation methods leads to non-smooth variation, and

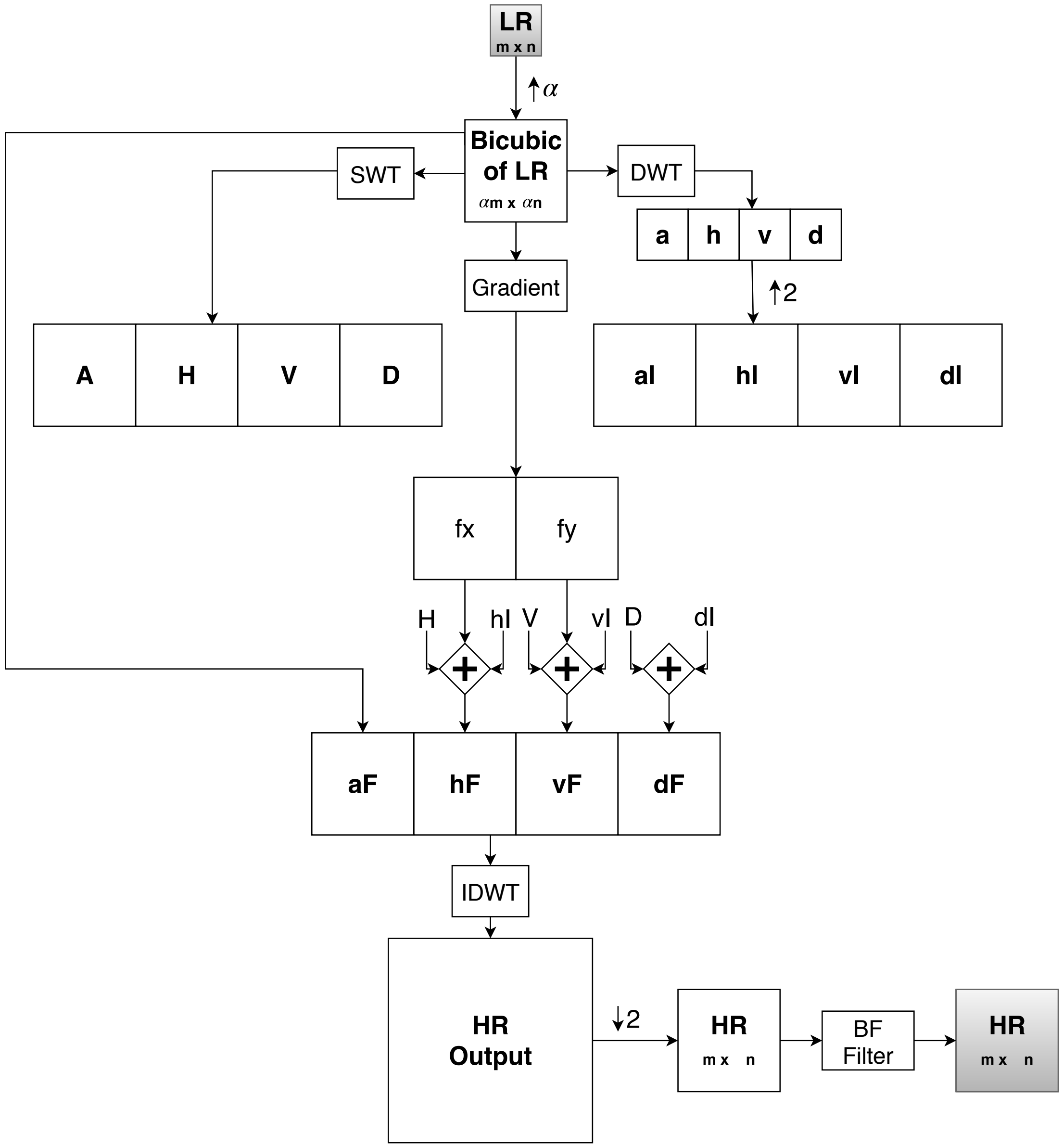


Figure 3.2: Block diagram of the proposed wavelet based method for single depth image super-resolution by factor α. The subbands A, H, V, D and a, h, v, d are the outcomes of one level SWT and DWT operation respectively. The subbands aI, hI, vI, dI are the interpolated versions of DWT outcomes, f_x, f_y are the gradients, and aF, hF, vF, dF are the final subbands for IDWT process.

higher order polynomial curve fitting like polynomial interpolation leads to over fitting the values. Hence, bicubic interpolation is used in the proposed method. Bicubic interpolation estimate the values of new data points from the known data points (neighboring 16 pixels) by fitting a smoother surface. It also involves a weighted averaging of nearby pixels, which yields some smoothing effect in the image. Such a smoothing will indeed help in noise reduction, but it also blurs the image details (e.g. edges). The output of bicubic interpolation method will be treated as an initial HR output and it is represented by D^0_{HR}. The next stage in the proposed wavelet based SR method will help in gaining the high-frequencies information.

3.2.2 High Frequency Combination

From the perspective of depth image super-resolution, edges are most important feature of depth image, unlike textures which is more prominent in optical images. Thus the essential task in depth image super-resolution is to preserve or improve these dominant edges while super-resolving the image. The high-frequency combination is an intermediate stage in the pipeline of the proposed method for improving the high-frequency information. This stage utilizes the contents obtained from DWT, SWT and gradients operation applied on bicubic interpolated image in the first state of the proposed method. The initially computed HR depth image estimate (D^0_{HR}) will be further improved by combining high-frequency content from these three image transformations.

A DWT operation is employed on D^0_{HR} which decompose the image into four subbands, i.e. low-low, low-high, high-low and high-high. The equations 3.1 and 3.2 shows the wavelet operation on the input image. Here the input image f is convolved with the basis ϕ to produce the wavelet output W.

$$W_\phi(j0, m, n) = \frac{1}{\sqrt{MN}} \sum_{x=1}^{M} \sum_{y=1}^{N} f(x, y)\phi_{j0,m,n}(x, y) \tag{3.1}$$

$$W^i_\psi(j, m, n) = \frac{1}{\sqrt{MN}} \sum_{x=1}^{M} \sum_{y=1}^{N} f(x, y)\phi^i_{j,m,n}(x, y) \tag{3.2}$$

where $f(\cdot)$ is the input image, and $\phi_{j,m,n}(x, y) = 2^{j/2}\phi(2^j x-m, 2^j y-n)$ and $\psi^i_{j,m,n}(x, y) = 2^{j/2}\psi^i(2^j x - m, 2^j y - n)$ are the wavelet basis functions with $i \in \{H, V, D\}$, and $W_\phi(j0, m, n)$ and $W^i_\psi(j, m, n)$ are the LL (low-low), LH (low-high), HL (high-low) and HH (high-high) subbands.

These subbands are half the resolution of D_{HR}^0, because of the DWT's implicit downsampling properties. The LL subband is an approximation band, and the other three subbands are the high-frequency components of the image in three different directions, viz. horizontal, vertical and diagonal.

One level DWT operation is shown in Figure 3.3 which produces four sub-bands. DWT decomposition applies low-pass and high-pass filters on the image and produces approximation, horizontal, vertical, and diagonal subbands. These subbands are also called coefficients named LL, LH, HL and HH, and these are represented by names a, h, v and d respectively. The dimensions of all of these subbands are downsampled to half as compared to the size of the input image. Based on the separable property of DWT, a 1D DWT can be applied along rows and columns separately. This DWT operation is applied on the initially estimated HR estimate D_{HR}^0 which produces four subbands a, h, v and d each of size $\frac{\alpha m}{2} \times \frac{\alpha n}{2}$. These subbands are further bicubicly interpolated by a factor of 2 to produce aI, hI, vI and dI each of size $\alpha m \times \alpha n$ which is equivalent to the size of D_{HR}^0.

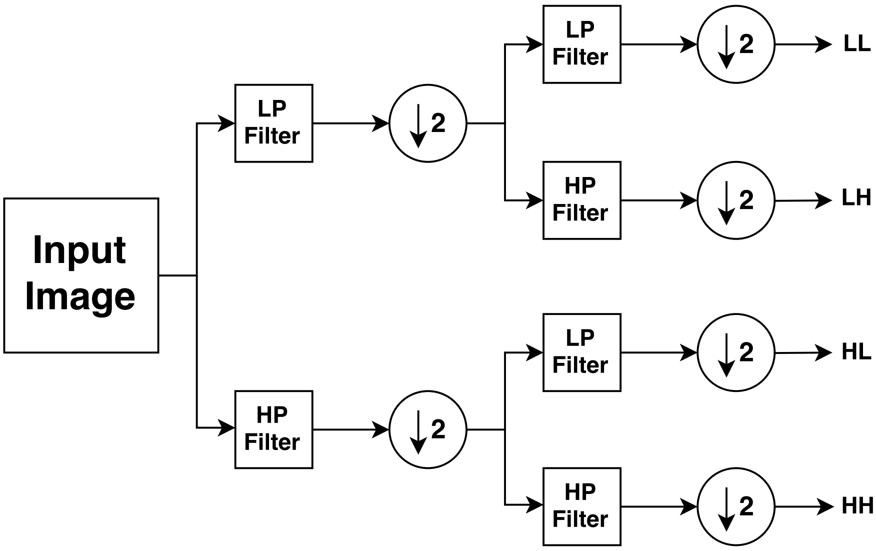


Figure 3.3: One level DWT operation on an image with low-pass (LP) and high-pass (HP) filter banks to produce four subbands.

In addition to DWT, the SWT operation is also performed on the same input D_{HR}^0. The SWT is designed to overcome the lack of translation-invariance of DWT due to downsamplers (choosing alternate rows and columns) and upsamplers. The upsampling is done by interpolating the LR image, where the interpolation is done via convolution with Haar wavelet coefficients. The SWT operation is similar to DWT operation except for the downsampling because the subbands (i.e. A, H, V and D) produced by SWT are of same size as the of the size of the input. The SWT is used to reduce the loss caused by DWT, as DWT downsamples the subbands which reduces the detailed information in an image, but SWT might suffer from memory constraints as compared to DWT.

For further improvement in the high-frequency content in the output image, the gradient information in horizontal and vertical direction (f_x and f_y) of the initial estimated HR image D_{HR}^0 is added to the horizontal and vertical subbands of the DWT and SWT respectively. The gradient of image I can be represented by Eq. 3.3.

$$\nabla I = \begin{bmatrix} f_x \\ f_y \end{bmatrix} = \begin{bmatrix} \frac{\partial I}{\partial x} \\ \frac{\partial I}{\partial y} \end{bmatrix} \tag{3.3}$$

where, the operator ∇ is the gradient operation applied on image I, and f_x and f_y are the two components in horizontal and vertical direction.

From Figure 3.2, the horizontal and vertical components, i.e. hF and vF, are the sum of interpolated horizontal and vertical components of DWT (i.e. hI and vI) with the horizontal and vertical components of SWT (i.e. H and V) respectively. Along with those two components, the hF and vF are further incremented with the calculated gradients f_x and f_y to get a sharper and cleaner image. Next, the diagonal components dI and D are added to produce dF, and finally the image D_{HR}^0 is considered as aF. All these four components will be fed to IDWT module discussed in following section.

Thus, the intermediate stage involves combining the high frequency information from DWT, SWT and gradient which will help in gaining the image details.

3.2.3 HR Image Estimation

In this stage, an inverse DWT (IDWT) operations has been used to reconstruct the output HR depth image. This stage takes four subbands (i.e. aF, hF, vF and dF) each of size $\alpha m \times \alpha n$, and produce an output image whose resolution is higher by a factor of 2 (i.e. $2\alpha m \times 2\alpha n$).

The IDWT operation on these four components gives output which is double than the required size for the upsampling factor α, thus, the final stage in the proposed method is the bicubic downsampler which downsamples the output of IDWT by a factor of 2 to get the desired HR image.

In literature, the SR methods which uses DWT (Demirel and Anbarjafari, 2011a) and SWT (Demirel and Anbarjafari, 2011b) apply wavelet transform operation directly on the input LR image itself. On the other hand, the proposed method employ the bicubic interpolation on the input LR image to produce an initial SR image D^0_{HR}, which has less noise and more image content. Thus, it is believed that the inputs to IDWT module (i.e. aF, hF, vF and dF), as shown in the block diagram in Figure 3.2, are relatively improved in terms of noise and image details.

3.2.4 Bilateral Filtering

While the bicubic interpolation can smooth some noise in the input image, it is not sufficient for heavy noisy in LR images. Hence, bilateral filter (BF) (Tomasi and Manduchi, 1998) has been used as a final stage to reduce the noise level thereby retaining the gained edges from previous steps. The choice of filter is important, as it is desired that the details recovered in the previous stage does not get affected by any filtering operation. Bilateral filter is known to perform well in this respect. The BF filter is a non-linear edge preserving smoothing filter which is defined as in Eq. 3.4,

$$\hat{I}(x) = \frac{1}{W_p} \sum_{x_i \in \Omega} I(x_i).f_s(\parallel x_i - x \parallel).f_r(\parallel I(x_i) - I(x) \parallel), \tag{3.4}$$

where I is the input image and $\hat{I}$ is the estimated noise free image, $f_s(\cdot)$ and $f_r(\cdot)$ are the spatial and range domain filter, Ω is the window size around pixel x, and W_p is the normalization factor. For an image I of size $m \times n$, the variance of the spatial filter σ_s is chosen as min$(m, n)/16$ and the variance for range filter σ_r is chosen as $0.1*$(max(I)-

min(I)). Filtering at a pixel is done by estimating the filter weights which depends on the spatial domain and range domain kernel. The final output will be super-resolved and noise free with, arguably, better details in the output HR depth image.

The pseudo code of the proposed method is shown in Algorithm 1 whose input is a low-resolution depth image D_{LR} and the output is the super-resolved depth image D_{HR} which is upsampled by $\times\alpha$ factor.

Algorithm 1 Pseudo code of the proposed method for depth image SR by factor α

```
INPUT: LR depth image $D_{LR}$ of size $m \times n$.
Initialize; $D^0_{HR}$=bicubic($D_{LR}$, $\alpha$)
Estimate gradient; $[f_x, f_y]$ = grad($D^0_{HR}$)
Apply DWT; $[a, h, v, d]$ = dwt_level1($D^0_{HR}$)
Apply SWT; $[A, H, V, D]$ = swt_level1($D^0_{HR}$)
Interpolate the DWT subbands; $[aI, hI, vI, dI]$ = bicubic_interp($a, h, v, d$)
Add the horizontal, vertical and diagonal subbands of DWT and SWT respectively
    to get $hTemp$, $vTemp$ and $dF$.
Add $hTemp$ and $vTemp$ with the gradients $f_x$ and $f_y$ of $D^0_{HR}$ respectively to get
    $hF$ and $vF$.
Apply IDWT; $[D^{temp}_{HR}]$ = idwt_level1($aF, hF, vF$ and $dF$)
Downsample the output $D^{temp}_{HR}$; $[D_{HR}]$ = downsample($D^{temp}_{HR}$).
if $D_{LR}$ was noisy then
    $D_{HR}$ = bilateral_filter($D_{HR}$)                ▷ only for noisy image
end if
OUTPUT: HR depth image $D_{HR}$ of size $\alpha m \times \alpha n$.
```

3.3 EXPERIMENTAL RESULTS AND DISCUSSIONS

This section present the SR results obtained from the proposed SRDWT method on several depth images taken from a popular depth image dataset of Middlebury (Scharstein and Szeliski, 2003), and its comparison with closely related DWT and SWT based SR methods (Demirel and Anbarjafari, 2011a,b). As the source code for these methods are not available for reproduction, it has been reimplemented to the best of the knowledge by utilizing all the information provided in Demirel and Anbarjafari (2011a,b). While these methods use *Daubechies* wavelet basis, it is reimplementated using *Haar* and *Daubechies* to maintain the consistency for comparisons purpose with the proposed

SRDWT method. Both the qualitative and quantitative results are shown, where PSNR and SSIM metrics are used to evaluate the performance of the SR methods. The results are demonstrated on both *noiseless* and *noisy* depth images.

The observed LR image is generated using the LR image model (Eq. 1.3) mentioned in Chapter 2. As mentioned earlier also, the LR image model is used only to generate the LR image, and it is no-where used in the reconstruction phase of the proposed SR method. From Eq. 1.3, the blurring filter $\mathcal{B}$ is a Gaussian filter with filter size of 7×7 and standard deviation of 1.6. The added noise is normally distributed with mean 0 and standard deviation 5. The experiments were performed with the popular wavelet basis of *haar* and *Daubechies*. Generally, *haar* is preferred in the case of SR problem, because it suites better for images with edges, whereas *Daubechies* have fixed filter values which makes it unreliable for SR problem.

The qualitative SR results for upsampling factor $\times 4$ are demonstrate on depth images *cones*, *art* and *reindeer*. The SR outputs shown in Figure 3.4, Figure 3.6 and Figure 3.8 are of *noiseless* images of *cones*, *art* and *reindeer* respectively. Similarly, the SR outputs shown in Figure 3.5, Figure 3.7 and Figure 3.9 are of *noisy* images of *cones*, *art* and *reindeer* respectively. The qualitative results shown in these figures are obtained using *haar* wavelet basis.

In all the figures mentioned above, the *top-row* shows the output image obtained from different SR methods against the GT image, and the *bottom-row* shows the cropped and zoomed portion of the respective images in *top-row*. It can be seen clearly that the SR output produced by the proposed SRDWT method looks sharper than the comparative methods from Demirel and Anbarjafari (2011a) and Demirel and Anbarjafari (2011b). As it can be observed (especially in the zoomed-in regions) that the proposed SRDWT method performs better than Demirel and Anbarjafari (2011a) and Demirel and Anbarjafari (2011b) in terms of less perturbations at edges.

For noisy images, it can be seen from the SR results that the SRDWT method performs well in retaining the depth edges, and in addition, it reduces the noise (thanks to the bicubic initial estimate and the bilateral filter). Which also mean that, the *haar* wavelet proves to be efficient for decomposing the image with edges as prominent features and used in wavelet based super-resolution task as it shows intrinsic relationship with super-resolution problems.

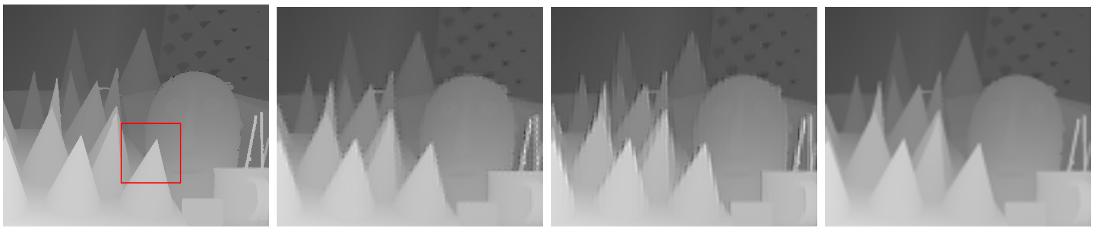
Table 3.1 and Table 3.2 shows PSNR and SSIM results for *noiseless* and *noisy* cases respectively. It shows the SR results for upsampling factors 2, 4, and 8. The average PSNR and SSIM has also been calculated over the set of chosen test images, it clearly shows that the proposed method outperforms DWT based SR method (Demirel and Anbarjafari, 2011a) and/or SWT based SR method (Demirel and Anbarjafari, 2011b).

Using *Daubechies* as wavelet basis, the proposed method perform 2.34 dB better than Demirel and Anbarjafari (2011a) and 5.89 dB better than Demirel and Anbarjafari (2011b), as opposed to using *Haar* wavelet basis for which the proposed method performs 2.25 dB and 3.84 dB better than Demirel and Anbarjafari (2011a) and Demirel and Anbarjafari (2011b) respectively for *noiseless* case for SR upsampling factor $\times 8$, and in fact the proposed method shows better results (more difference) at higher upsampling factors. Note that the improvement is consistent for all upsampling factors. It is also noted that the results with the *Haar* wavelet basis performs better than the *Daubechies* wavelet basis for all approaches.

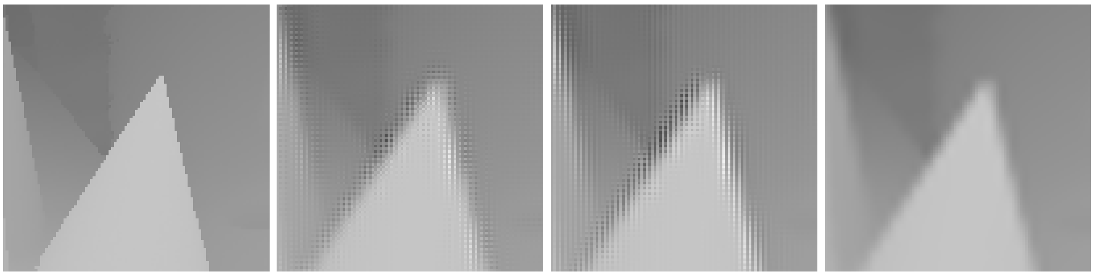
The competence of the proposed SRDWT method can be seen in the graph shown in Figure 3.10(a) and Figure 3.10(b) on *noiseless* and *noisy* images respectively. The graph shows the average PSNR value computed from Table 3.2 and Table 3.2 over all the test images chosen for experimentation.

3.4 SUMMARY

The proposed wavelet transform based single depth image super-resolution is simple yet effective method. It improves the high-frequency content of output image via enhancing the high-frequency subbands obtained from DWT and SWT operation on input LR image, which is then combined with the gradient information extracted from the input LR image. The proposed method is divided into four stages. The initial bicubic

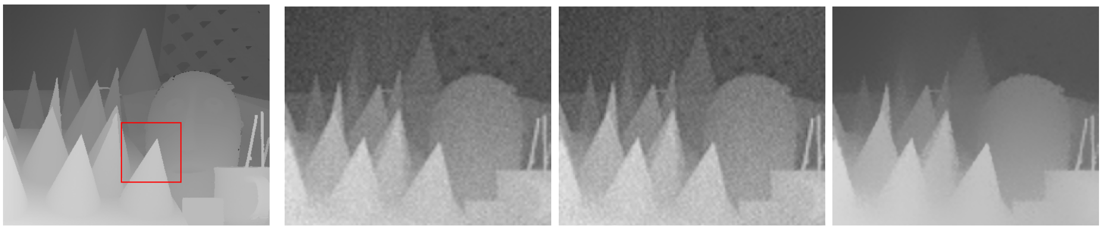


(a) GT (b) DWT (c) SWT (d) Proposed

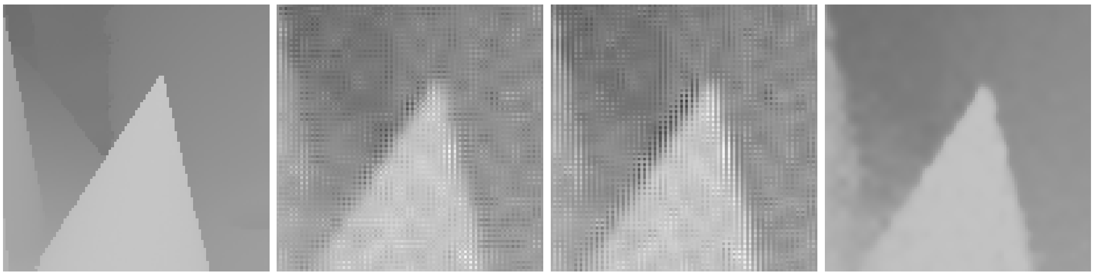


(e) (f) (g) (h)

Figure 3.4: SR results comparison for upsampling factor ×4 on *noiseless* ($\sigma = 0$) depth images *Cones*. **Top row**: Ground truth, DWT (Demirel and Anbarjafari, 2011a), SWT (Demirel and Anbarjafari, 2011b), Proposed. **Bottom row**: zoomed region of the above images respectively.

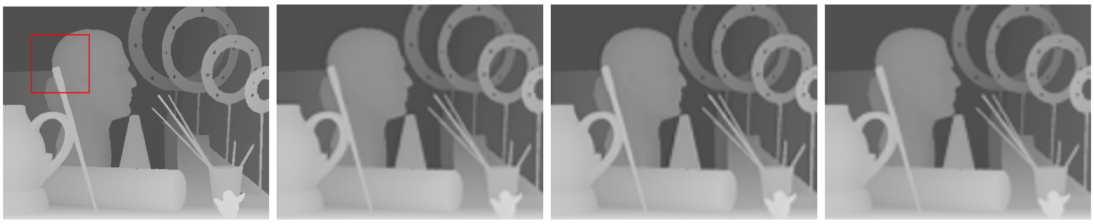


(a) GT (b) DWT (c) SWT (d) Proposed

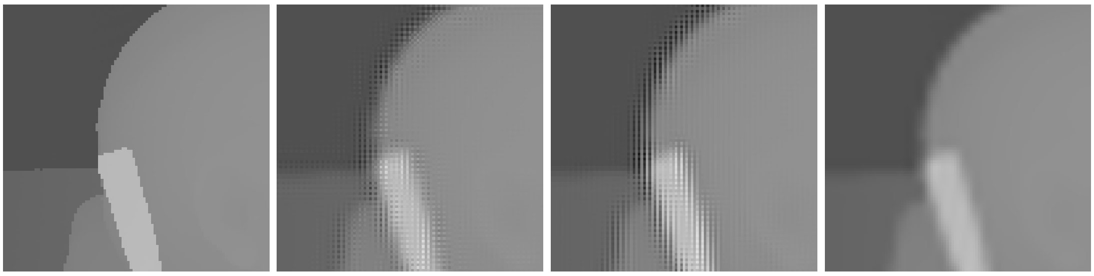


(e) (f) (g) (h)

Figure 3.5: SR results comparison for upsampling factor ×4 on *noisy* ($\sigma = 5$) depth images *Cones*. **Top row**: Ground truth, DWT (Demirel and Anbarjafari, 2011a), SWT (Demirel and Anbarjafari, 2011b), Proposed. **Bottom row**: zoomed region of the above images respectively

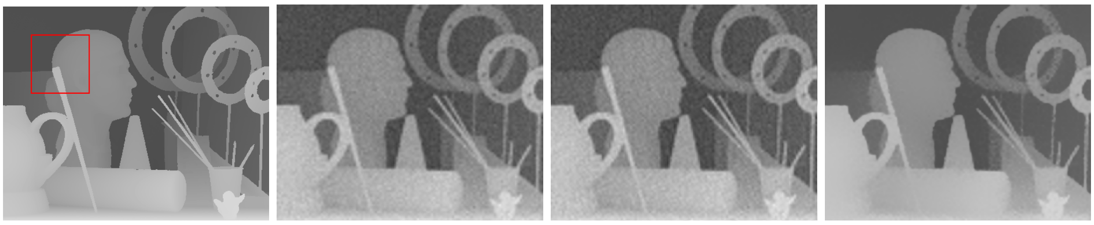


(a) GT (b) DWT (c) DWT-SWT (d) Proposed

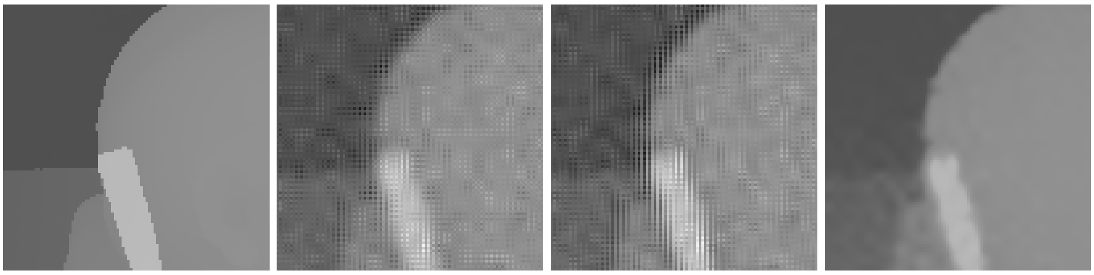


(e) (f) (g) (h)

Figure 3.6: SR results comparison for upsampling factor ×4 on *noiseless* ($\sigma = 0$) depth images *Art*. **Top row**: Ground truth, DWT (Demirel and Anbarjafari, 2011a), SWT (Demirel and Anbarjafari, 2011b), Proposed. **Bottom row**: zoomed region of the above images respectively.

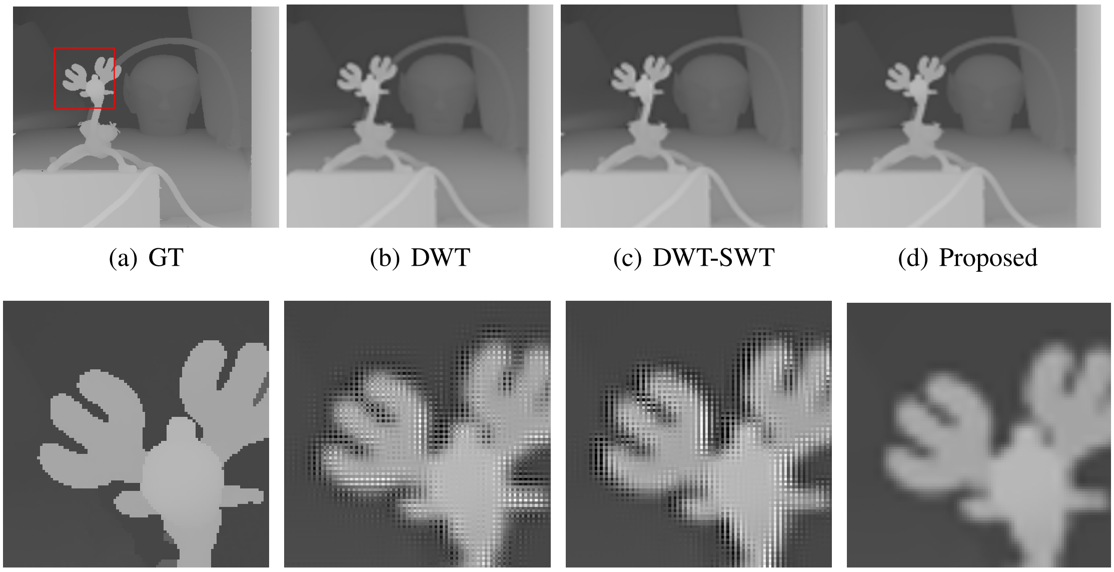


(a) GT (b) DWT (c) SWT (d) Proposed



(e) (f) (g) (h)

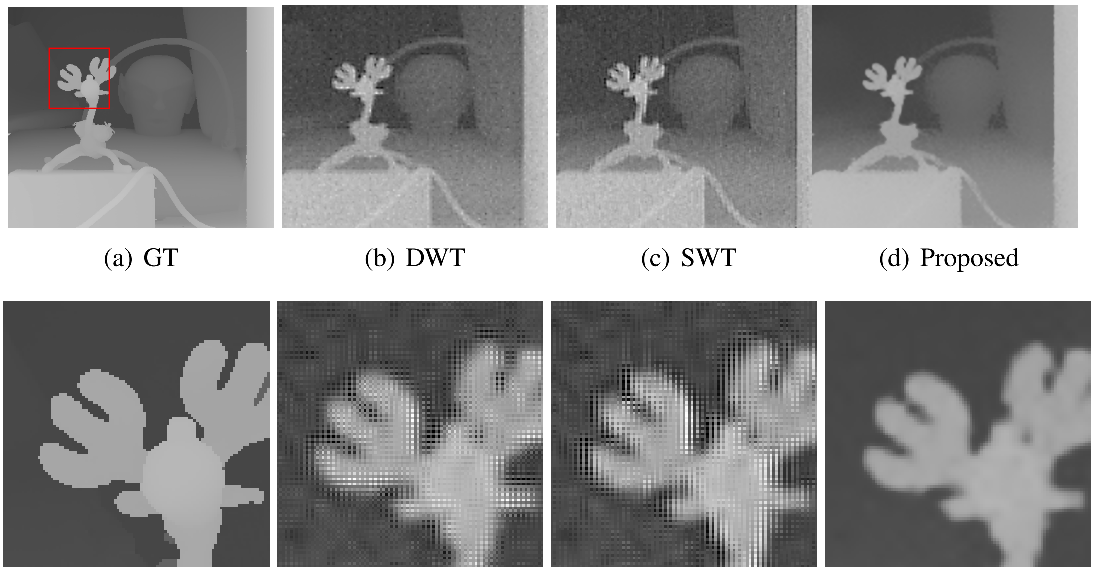
Figure 3.7: SR results comparison for upsampling factor ×4 on *noisy* ($\sigma = 5$) depth images *Art*. **Top row**: Ground truth, DWT (Demirel and Anbarjafari, 2011a), SWT (Demirel and Anbarjafari, 2011b), Proposed. **Bottom row**: zoomed region of the above images respectively



(a) GT (b) DWT (c) DWT-SWT (d) Proposed

(e) (f) (g) (h)

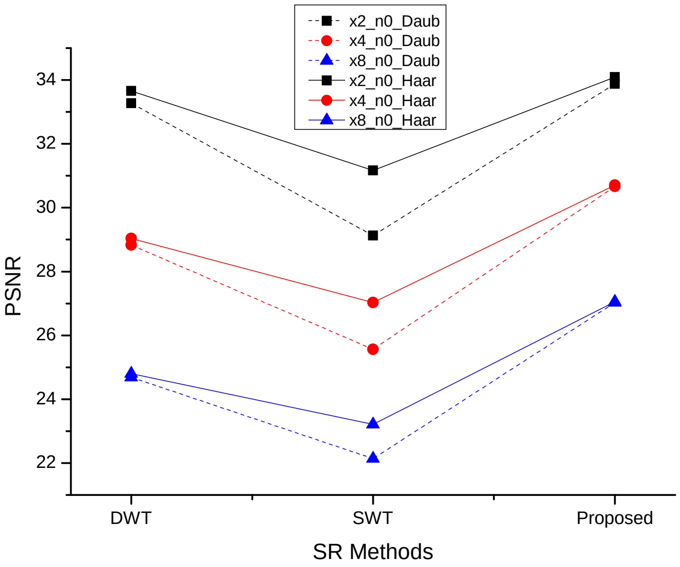
Figure 3.8: SR results comparison for upsampling factor $\times 4$ on *noiseless* ($\sigma = 0$) depth images *Reindeer*. **Top row**: Ground truth, DWT (Demirel and Anbarjafari, 2011a), SWT (Demirel and Anbarjafari, 2011b), Proposed. **Bottom row**: zoomed region of the above images respectively



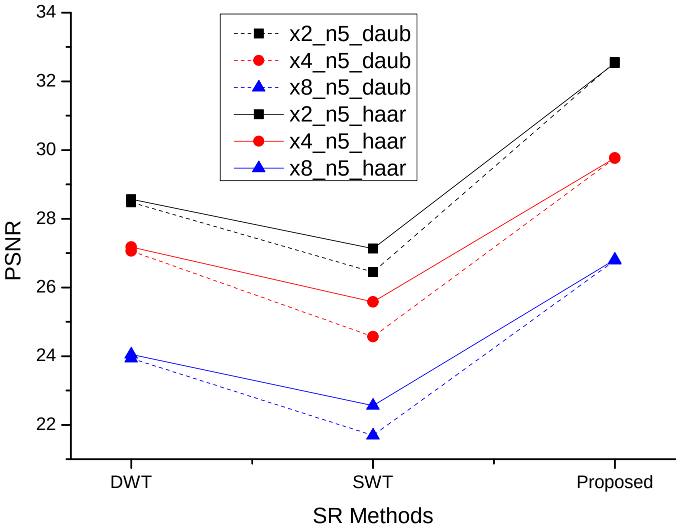
(a) GT (b) DWT (c) SWT (d) Proposed

(e) (f) (g) (h)

Figure 3.9: SR results comparison for upsampling factor $\times 4$ on *noisy* ($\sigma = 5$) depth images *Reindeer*. **Top row**: Ground truth, DWT (Demirel and Anbarjafari, 2011a), SWT (Demirel and Anbarjafari, 2011b), Proposed. **Bottom row**: zoomed region of the above images respectively



(a)



(b)

Figure 3.10: Average PSNR result comparison of proposed SRDWT method with other SR methods for several upsampling factors on both *noiseless* (**left**) and *noisy* (**right**) images. Notation $\times i\_nj\_basis$ indicates upsampling factor i, noise standard deviation j, and wavelet basis function $basis$.

Table 3.1: PSNR/SSIM comparison of SR by factor 2, 4 and 8 on *noiseless* depth images ($\sigma = 0$)

Factor	Images	Using Daubechies Wavelet Basis			Using Haar Wavelet Basis		
		DWT	SWT	Proposed	DWT	SWT	Proposed
×2	Cones	35.53/0.96	30.37/0.94	**36.44/0.97**	36.22/0.96	33.50/0.94	**36.63/0.97**
	Art	30.75/0.91	27.35/0.87	**31.25/0.92**	31.04/0.91	28.85/0.88	**31.49/0.93**
	Reindeer	33.57/0.95	28.88/0.93	**34.30/0.97**	33.97/0.96	30.53/0.94	**34.51/0.97**
	Aloe	33.21/0.94	29.90/0.91	**33.54/0.95**	33.41/0.95	31.79/0.93	**33.73/0.96**
Average		33.27/0.94	29.13/0.91	**33.88/0.95**	33.66/0.95	31.17/0.92	**34.09/0.96**
×4	Cones	30.88/0.89	26.96/0.86	**33.06/0.95**	31.38/0.91	29.28/0.87	**33.08/0.95**
	Art	26.34/0.76	23.77/0.72	**28.40/0.88**	26.47/0.78	24.77/0.75	**28.46/0.88**
	Reindeer	29.39/0.88	25.46/0.83	**31.16/0.94**	29.60/0.89	26.78/0.86	**31.19/0.94**
	Aloe	28.71/0.86	26.04/0.81	**30.07/0.92**	28.69/0.86	27.30/0.84	**30.12/0.92**
Average		28.83/0.85	25.56/0.81	**30.67/0.92**	29.04/0.86	27.03/0.83	**30.71/0.92**
×8	Cones	27.05/0.77	23.78/0.72	**29.57/0.92**	27.32/0.80	25.57/0.76	**29.58/0.92**
	Art	22.54/0.56	20.33/0.50	**24.75/0.81**	22.49/0.59	20.89/0.57	**24.76/0.81**
	Reindeer	25.11/0.73	21.96/0.66	**27.62/0.91**	25.30/0.77	23.22/0.75	**27.65/0.91**
	Aloe	24.06/0.71	22.49/0.66	**26.19/0.87**	24.07/0.73	23.14/0.73	**26.22/0.87**
Average		24.69/0.69	22.14/0.64	**27.03/0.88**	24.80/0.72	23.21/0.70	**27.05/0.88**

Table 3.2: PSNR/SSIM comparison of SR by factor 2, 4 and 8 on *noisy* depth images ($\sigma = 5$)

Factor	Images	Using Daubechies Wavelet Basis			Using Haar Wavelet Basis		
		DWT	SWT	Proposed	DWT	SWT	Proposed
×2	Cones	29.24/0.57	27.16/0.54	**34.29/0.95**	29.36/0.58	28.04/0.53	**34.22/0.95**
	Art	27.50/0.55	25.40/0.50	**30.67/0.92**	27.60/0.56	26.07/0.50	**30.65/0.91**
	Reindeer	28.61/0.55	26.34/0.51	**33.25/0.95**	28.71/0.55	26.93/0.50	**33.22/0.95**
	Aloe	28.57/0.57	26.90/0.53	**32.04/0.94**	28.61/0.58	27.46/0.52	**32.01/0.93**
Average		28.48/0.56	26.45/0.52	**32.56/0.94**	28.57/0.57	27.13/0.51	**32.53/0.94**
×4	Cones	28.47/0.65	25.74/0.61	**31.95/0.94**	28.72/0.66	27.14/0.61	**31.95/0.94**
	Art	25.25/0.57	23.07/0.51	**27.70/0.87**	25.36/0.58	23.89/0.52	**27.73/0.87**
	Reindeer	27.47/0.63	24.53/0.57	**30.20/0.93**	27.61/0.64	25.45/0.58	**30.18/0.93**
	Aloe	27.03/0.63	24.95/0.57	**29.17/0.91**	27.02/0.63	25.82/0.59	**29.20/0.91**
Average		27.06/0.62	24.57/0.57	**29.76/0.91**	27.18/0.63	25.58/0.58	**29.77/0.91**
×8	Cones	25.94/0.62	23.22/0.55	**29.39/0.92**	26.19/0.64	24.60/0.58	**29.41/0.92**
	Art	22.03/0.46	20.00/0.39	**24.50/0.81**	22.01/0.48	20.52/0.43	**24.52/0.81**
	Reindeer	24.36/0.58	21.58/0.50	**27.25/0.91**	24.50/0.60	22.60/0.54	**27.27/0.91**
	Aloe	23.41/0.57	21.97/0.50	**26.00/0.87**	23.48/0.58	22.50/0.54	**26.02/0.87**
Average		23.94/0.56	21.69/0.49	**26.79/0.88**	24.05/0.58	22.56/0.52	**26.81/0.88**

estimation stage helps in providing better content to the high-frequency enhancement stage and some noise robustness. The bilateral filter in the final stage also helps in noise reduction while preserving the edges enhanced by the intermediate stages. Various experiments conducted on depth images from Middlebury dataset demonstrate the potential of the proposed method in performing the super-resolution task on noiseless and noisy cases. The proposed method has been compared with some related DWT and SWT based super-resolution methods, and the proposed method found to be much superior amongst all.

It is realized from the experiments that wavelet based method will distort the SR output for higher upsampling factors. It has motivated to make use of some extra source of information to boost the SR results. In the next contributory chapter it is shown that how an input LR image along with the HR guidance image is used for super-resolution. The idea of combining the LR depth image and HR colour image is in existence, and lately it can be seen in set up like Microsoft Kinect where both the depth camera and the optical cameras are mounted on same rig.

CHAPTER 4

DEPTH IMAGE SUPER-RESOLUTION USING HR GUIDANCE COLOUR IMAGE

4.1 INTRODUCTION

[1] Commercially available modern time-of-flight (ToF) depth cameras cannot obtain high-resolution depth images. The images captured by these cameras generally suffer from lower spatial resolution, noise and missing regions. On the other hand, the sophisticated depth cameras could capture high-resolution depth images, but they have intensive capturing time which makes it unsuitable for real-time applications.

In literature, there are methods which makes use of some guidance image to improve the resolution of the LR depth image. The guidance image can be any other high-resolution image so that the SR method can be able to extract some information from it and super-resolve the LR depth image to have sharp edges. The guidance image can be in the form of gray image, colour image, or the depth image itself. Several depth image SR methods have used HR guidance colour image because obtaining the depth image along with the colour image is easy with current camera technology. In many cases, the depth cameras are integrated with the optical cameras on a same rig e.g. Microsoft Kinect, so that it can readily capture LR depth image and HR colour image of the same scene. Most of the cameras of such construction produces registered images, however, there can be cameras which does not have inbuilt image registration module, in that case these images (LR depth image and HR colour image) need to be registered externally.

The proposed method described here belongs to the class of SR methods which utilize a corresponding registered HR color image of the same scene. The overview

---

[1] Chandra Shaker Balure, M. Ramesh Kini, and Arnav Bhavsar. ”Depth Image Super-resolution with Local Medians and Bilateral Filtering.” *Eleventh International Conference on Industrial and Information Systems (ICIIS-2016)* IEEE, 2016.

of the HR guidance colour image based depth image SR is shown in Figure 4.1. Such methods assume that the edges in the colour image coincide with the edges in the depth image. This assumption is valid as most of the camera setup comes with two cameras (depth camera + optical camera) on a same rig and they are capable of producing a co-planar images where the dominant edges of both images (depth image and optical image) coincides.

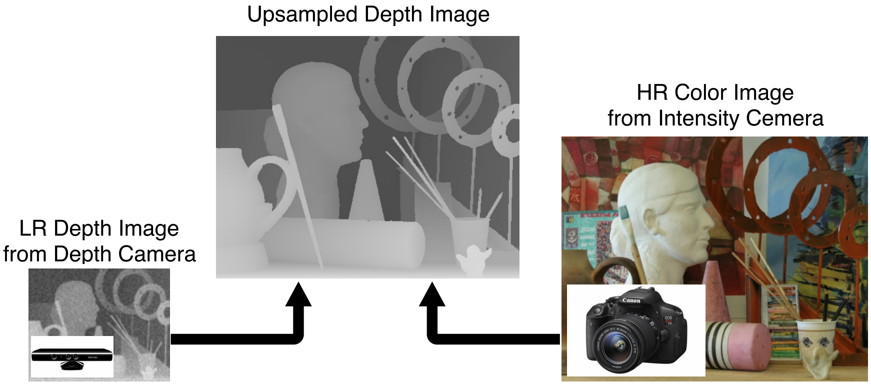


Figure 4.1: Depth image upsampling aided by the HR color image.

The HR guidance colour image for depth image SR is preferred mostly over multiple image based depth SR methods or training example based depth SR methods. The reason is that, in multiple image based depth SR methods more number of LR images are required to find the non-redundant information from each one of the LR image to fuse it into the final HR image. Similarly, in training example based depth SR methods a large training dataset is required to learn the mapping of HR-LR intrinsic features such that the learned mapping can correctly estimate the corresponding HR patch for a never seen LR patch.

In guided image based methods for SR, the HR colour image is the second input along with the LR depth image. The assumption is that both LR depth image and HR colour image are registered with each other. It means that they are co-aligned with each other such that the prominent edges in the depth image coincide with the edges of the colour image. The correspondence between these two images gives more advantages to learn the prior (e.g. edge prior, gradient prior, etc.).

The proposed SR method presented here is a simple yet effective method which use segmentation of the HR guidance color image to represent such discontinuities and locally smooth regions. As compared to other guidance based SR methods, the proposed SR methods has simple local operations and is quite efficient. Note that, the crux of the proposed method is in the explicit consideration of segments rather than filtering, which enforces to produce sharper edge regions. While it does produce some artifacts near the discontinuities, the use of bilateral filtering is used to mitigate such artifacts. Note that the use of filtering is only used as post processing. The results are demonstrated for noiseless and noisy images, which is typically not considered in existing guidance based SR methods (except in NAFDU method (Chan et al., 2008)).

The proposed HR guidance colour image based SR method is demonstrated with two different variants. The first variant method is called LRBicSR, and the second variant method is called LRSR. In a nutshell, the LRBicSR method takes an input LR depth image, which is then bicubically interpolated to the resolution equivalent to the resolution of the guidance image. It is treated as an initial estimate for the proposed SR method. On the other hand, LRSR method takes an LR depth image and it is mapped on the HR grid by uniformly placing the LR depth points.

Further, the segments in the HR guidance color image are computed using popular segmentation method i.e. mean shift algorithm (Comaniciu and Meer, 2002) and simple linear iterative clustering (SLIC) (Achanta et al., 2012). These segments are used as a cue to super-resolve the LR depth image to the resolution equivalent to that of the HR colour image. Corresponding to each segment of the HR color image, the depth values for the SR image are computed based on computation of local medians and the values from interpolated image or the plane fitting approach. While this process yields crisp edges in the SR depth image, it can also result in some artifacts at the abutting regions of the segments. To reduce such artifacts, bilateral filter (BF) (Tomasi and Manduchi, 1998) is employed, which is an edge preserving smoothing filter.

It is demonstrated that the proposed guidance based SR method is able to achieve good localization even at higher upsampling factors (e.g. $\times 4$ and $\times 8$). Interestingly, it is also demonstrated that such a relatively simplistic approach involving segmentation, median estimation, and filtering can also performs well under noisy cases. Moreover, some variants of the proposed method depending on segmentation methodology (either MS or SLIC) used for segmentation, and the presence or absence of bilateral filtering, has also been shown.

4.2 SEGMENTATION ALGORITHMS

This section presents a brief discussion of the foundation process in the proposed method i.e. image segmentation. Segmentation of colour image is the process of partitioning the image into well defined segments. Segmentation can be done in many ways, as there exist many approaches, e.g. region growing segmentation, k-means clustering, mean-shift (MS) segmentation method, simple linear iterative clustering (SLIC) segmentation. The segmentation method used in the proposed HR guidance based SR method is the existing popular segmentation approach i.e. mean-shift (MS) segmentation approach and simple linear iterative clustering (SLIC) segmentation approach. These segmentation methods provide a prior information for the SR problem.

Mean-Shift Segmentation Algorithm

The MS segmentation method is a low-level vision tasks, e.g. discontinuity preserving smoothing, and image segmentation. The MS algorithm is a density estimation-based non-parametric clustering approach, where, the colour image is converted into L*u*v colour space. In the transformed colour space, MS algorithm tries to estimate the *modes* of the unknown density, and then it clusters the region which is close to the mode density based on the local structure.

Kernel density estimation (also know as Parzen window technique) is widely used density estimation method. For a given n input points $x_i \in \mathbb{R}^d$, where, $i = 1, \cdots, n$, the multivariate kernel density estimator at a point x, with kernel $K(x)$, and symmetric positive definite $d \times d$ bandwidth matrix H is given by,

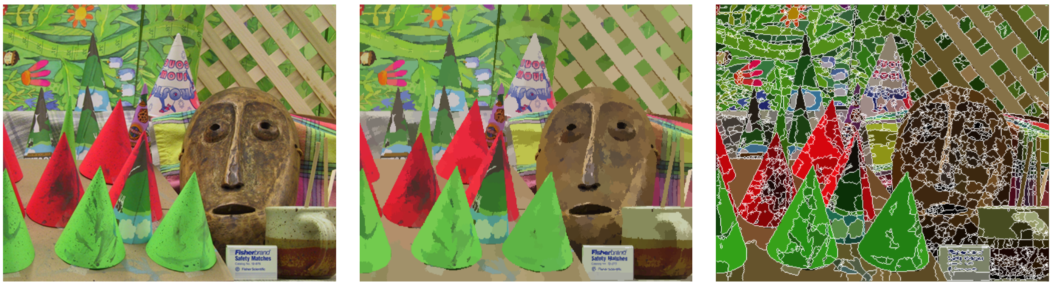
$$\hat{f}(x) = \frac{1}{n} \sum_{i=1}^{n} K_H(x - x_i), \tag{4.1}$$

The zeros gradient ($\nabla f(x) = 0$) denotes the mode location, and hence, MS approach is an elegant way to locate these zeros without estimating the density. The MS algorithm iteratively performs computation of mean shift vector and translation of kernel window till it converges to a point which has zero gradient. Initially, all the points are treated as cluster center (or modes), and after the repeated computation and translation steps, it converges to the mode point of the density. The Epanechnikov kernel is used, and the mean shift vector will always point to dense region. On initialization, the mean shift vector of all the points start to drift towards the maximum increase in the density. As the kernel window reaches local maxima, the step size taken are small. It can also be considered as an adaptive gradient ascent method. At the end of the process, all the colour regions are grouped separately into a segment. In MS procedure, the critical part is the selection of bandwidth parameter, i.e. spatial bandwidth (h_s), and range bandwidth (h_r).

The core steps of MS algorithm is as follows:

1. Define a kernel window around each data point in d-dimensional space R^d.
2. Compute the mean of the data points in each window.
3. Compare the old and the new mean values. If the difference is greater than specified convergence threshold, then shift the window to the new center by the amount computed by mean shift vector.
4. Repeat step 3 until convergence.

Figure 4.2 shows the result of colour image segmentation of *cones* image from Middlebury dataset (Scharstein and Szeliski, 2002), where first image is the original colour image, the middle image is the segmented image, and the last image is the segmented image with region boundary delineated.



(a) Original image (b) MS segmented image (c) MS segmented image with region boundary delineated

Figure 4.2: MS segmentation results on *cones* colour image

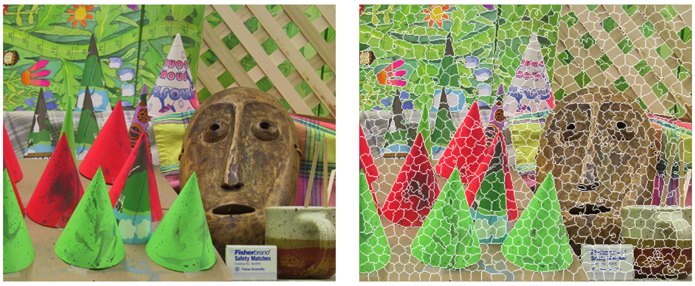
Simple Linear Iterative Clustering Segmentation Algorithm

Other than the MS segmentation method the SLIC segmentation (Achanta et al., 2012) is another method which has been used as alternate method for segmenting the colour image. SLIC require only one parameter, k, which decides the number of superpixels (or segments) required to segment the image. SLIC convert the colour image into CIELAB colour space to segment the colour image. The distance metric in CIELAB space is non-trivial. The colour of a pixel is represented by $[l\ a\ b]^T$, and its position is represented by $[x\ y]^T$ in the CIELAB colour space, thus the conventional euclidean distance in $labxy$ does not work well for different superpixel sizes. Thus, the two distances (spatial proximity, and colour proximity) are combine into single measure, which can be represented as:

$$D' = \sqrt{\left(\frac{d_c}{N_c}\right)^2 + \left(\frac{d_s}{N_s}\right)^2} \tag{4.2}$$

It initialize cluster centers on a regular grid which is S pixels apart, and the grid interval is $S = \sqrt{N/k}$, where, N is the total number of pixels in the image which is to be segmented. It uses gradient ascent to iteratively refine the clusters until the convergence criterion is met to form the required number of super-pixels. Figure 4.3 shows the results of SLIC segmentation on *cones* colour images.

It can be seen that both MS and SLIC segmentation approaches segments the image such that the pixels across the object boundaries most likely fall into different segments, which is essential. This is important for the depth enhancement (either DR or SR ap-



(a) Original image (b) SLIC segmented image with region boundary delineated

Figure 4.3: SLIC segmentation results on *cones* colour image

proaches) to have depth discontinuities (which are essentially at the prominent object boundaries) coincide with the segment boundaries.

4.3 PROPOSED HR GUIDANCE IMAGE BASED SR METHOD *WITH* INITIAL BICUBIC ESTIMATE

This section presents one of the variants of the proposed HR guidance image based SR method, i.e. LRBicSR. The whole process of LRBicSR method is shown in Figure 4.4. There are two inputs, one is the LR depth image (D_{LR}) and the second one is its corresponding HR colour image (C_{HR}). The LR depth image is first interpolated using bicubic technique to produce D_{Interp}. The MS or SLIC segmentation method is applied on C_{HR} to obtain C_{seg}. The segments in C_{seg} and bicubic values in D_{Interp} are combined in an intelligent way to fill the segment region in the desired HR grid D_{SR}. The dimension of each of these input images, intermediate images and the final image are represented as $D_{LR} \in \mathbb{R}^{m \times n}$, $C_{HR}, C_{seg}, D_{Interp}, D_{SR} \in \mathbb{R}^{\alpha m \times \alpha n}$. The whole process of LRBicSR method is divided into four stages, i.e. initial estimate, colour image segmentation, HR depth image estimation, and bilateral filtering, each of which is discussed in detail in the following sections.

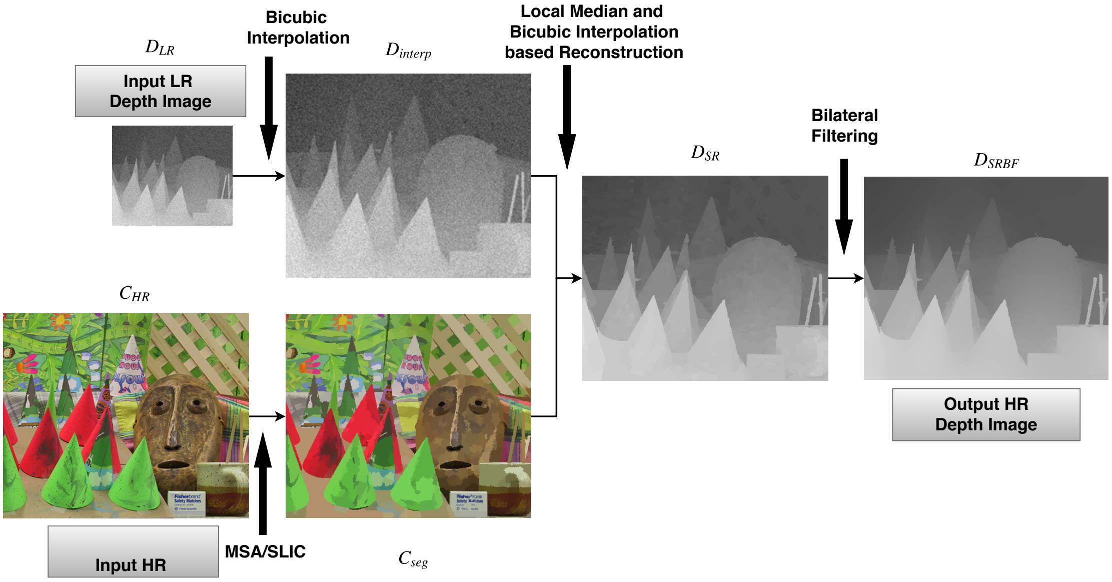


Figure 4.4: Block diagram of the proposed approach for depth image super-resolution.

4.3.1 Initial Estimate

The proposed LRBicSR method presented here starts with bicubically interpolating the LR depth image to the dimension equal to that of the HR color image (or that of the desired SR depth image). While bicubic interpolation can smooth the edges, it preserves the overall shape, and can serve as a good initial estimate for super-resolution. As depth images are largely texture less, the depth values can also be directly borrowed from the interpolated image for reconstructing smooth regions in the SR image (except edge regions), as it is discusses in following section. In addition, the interpolation process involves smoothing as well as noise reduction in case of noisy LR depth images.

4.3.2 HR Colour Image Segmentation

For segmenting the HR colour image, well known segmentation methods have been used, i.e. mean-shift segmentation method (MS) (Comaniciu and Meer, 2002) or simple linear iterative clustering (SLIC) (Achanta et al., 2012) method.

Employing MS segmentation on colour image produces local segments which are edge aligned. These can be used as a cue for super-resolving the depth images by considering the LR interpolated depth image.

Alternative to MS segmentation, SLIC segmentation method (Achanta et al., 2012) has also been used. It require only one parameter to set, k, which indicates the number of superpixels (or segments) required to segment the image. It initializes each cluster center on a regular grid which is S pixels apart with grid interval $S = \sqrt{N/k}$, where N is the total number of pixels in the image. It uses gradient ascent to iteratively refine the clusters until the convergence criterion is met to form the required number of superpixels.

From these segmentation methods, the local segments are obtained which are used as cues to assist in estimating the unknown depth values in the output SR depth image.

4.3.3 SR Image Estimation

The SR image estimation stage uses segmented colour image and bicubic interpolated depth image, as shown in Figure 4.4. Both the images have same spatial resolution. In the proposed LRBicSR method, for each segment in the HR color image C_{seg}, the corresponding segment region is co-located in the interpolated depth image D_{Interp}. A difference of the maximum and the minimum depth value in the co-located local segment of D_{Interp} is computed, as shown in Eq. 4.3,

$$d^{s}_{diff} = \max(D_{Interp}(s)) - \min(D_{Interp}(s)) \tag{4.3}$$

where s denotes the segment with sets of pixel locations, and $D_{Interp}(s)$ denotes the set of pixel values in a segment s of D_{Interp}, and d^{s}_{diff} denotes the different between the max and min depth value for the segment s. The segment region in D_{Interp} is reconstructed either by taking the interpolated values from the local segment or the estimated median value of the all the pixels of that local depth segment, which is decided by the threshold value τ, which is mathematically shown in Eq. 4.4,

$$\begin{aligned} D_{SR}(s) &= D_{Interp}(s) \text{ if } d_{diff} < \tau \\ &= \text{median}(D_{Interp}(s)) \text{ if } d^s_{diff} \geq \tau \end{aligned} \tag{4.4}$$

The case where $d_{diff} < \tau$ indicates that the region corresponding to the segment s is relatively smooth (not near edges), and thus, copying the bicubic information from the interpolated depth image is beneficial. On the other hand, $d_{diff} \geq \tau$ indicates that a region is either noisy or is close to an edge. As the edge pixels in the interpolated images are smoothed out, such pixels are included in segments which are nearby edges. Moreover, in case of noisy images, regions of gradual depth variation also yield a large d_{diff}. In such a case, a local median values is used to fill the SR depth image near such regions. Note that, in this case, a constant value is assigned to a segment in D_{SR}. However, as the image is over-segmented, the segment size become small, and a constant depth assumption proves to be a good approximation in such local regions. This indeed helps us to mitigate the smoothing near the edges as present in the bicubic interpolated image, and also greatly helps in noise reduction.

The point to note here is that, copying the bicubic values at flat region does not effect much in the HR output, but if such approach of copying bicubic values at edge regions is employed then it might degrade the HR output quality heavily. However, using median values at edge region gives sharper HR output image, but at the cost of some artifacts if the segment region is not properly aligned with the edges in the HR colour image. If the similar technique is used for noisy images, then the HR output will be noisy, as because, in flat regions in noisy depth images, copying bicubic will copy the noise also, which leads to unnecessary degradation in the HR output quality. Hence, a reverse technique is used, where, the median value is used at the flat regions in the depth image, and bicubic values are used at edge regions.

It has been experimentally observed that, for the values of τ that is used, on an average around 75% of the total number of segments use bicubic values, and 25% of it use median, which means the majority of the segments are smoother, and copying the bicubic values from the interpolated LR depth image into the output HR depth image gives smoother effect (which is what is expected at the smooth regions), and the remaining regions are filled with the median value of their corresponding segment regions of interpolated LR depth image. A similar analysis is done for noisy depth images, where on an average around 95% of segments are flat, and 5% segments are edgy. Hence the above concept is reversed, and fill the majority of the segment regions in the super-resolved depth image with the median value to have a smoother HR depth output. The percentage of flat and edgy segments are based on the threshold value τ details of which can be seen in the results section.

4.3.4 Bilateral Filtering

While the use of local medians yields sharper discontinuity localization in the SR output image, it is sometimes accompanied by some artifacts. This is due to the fact that the median computation involves pixels of the blurred edges from the interpolated depth image. To further mitigate such artifacts, a bilateral filter (BF) (Tomasi and Manduchi, 1998) is employed as a final stage in the proposed SR pipeline. Importantly, as BF filter is an edge preserving smoothing filter, it helps in reducing noise/artifacts, but by largely preserving the edge information, unlike the Gaussian smoothing filter which blurs all the image details. Mathematically, BF filter can be represented as shown earlier in Eq. 3.4, where I is the input image and $\hat{I}$ is the estimated noise free image, $f_s(\cdot)$ and $f_r(\cdot)$ are the spatial and range domain filter, Ω is the window size around pixel x, and W_p is the normalization factor. BF filtering also helps in noisy scenarios by reducing residual noise in the SR depth image.

The overall proposed SR method is summarized in the pseudo code shown in Algorithm 2.

Algorithm 2 Pseudo code for guidance depth image SR using MSA segment cues

```
**Data:** LR depth image $D_{LR}$, HR colour image $C_{HR}$
**Result:** SR depth image $D_{SR}$
Initialize $D_{SR} = 0$
$D_{Interp}$ = bicubic_interp($D_{LR}$)
[$C_{seg}, L$]= mean_shift($C_{HR}$)
for $i = L(1) : L(end)$ do
    $reg$ = extract_region($C_{seg}$)$^{L(i)}$
    $valVector$ = $D_{Interp}(reg)$
    $d_{diff}$ = max($valVector$) - min($valVector$)
    if $d_{diff} < \tau$ then
        $D_{SR}(reg)$ = $D_{Interp}(reg)$
    else
        $D_{SR}(reg)$ = median($valVector$)
    end if
end for
$D_{SR}$ = bilateral_filter($D_{SR}$)
```

4.4 PROPOSED SR GUIDANCE BASED SR METHOD *WITHOUT* INITIAL ESTIMATE

This section presents LRSR method based on the proposed HR guidance colour image based SR technique. Here, the LR image is mapped on to the HR grid of the required spatial resolution. The SR method with such a HR grid input where the depth values are missing at alternate locations (for $\times 2$) can be treated as depth reconstruction problem.

Figure 4.5 shows the block diagram of LRSR method, where the inputs are similar to that of LRBicSR method, i.e. LR depth image and the corresponding registered HR colour image. Unlike LRBicSR method, in LRSR method we do not perform bicubic interpolation to achieve the initial HR image, but instead the LR depth points are mapped uniformly on the HR grid. This HR grid has a resolution equal to the the required spatial resolution of the SR output. Here, the LRSR problem is approached in two ways i.e. *directly* and *hierarchically*, where the later approach is especially for higher upsampling factors. Both these approaches can be seen as depth reconstruction problem where a dense depth reconstructed output is obtained from a sparse HR grid.

In direct approach of super-resolution, LR image points are laid on the HR target

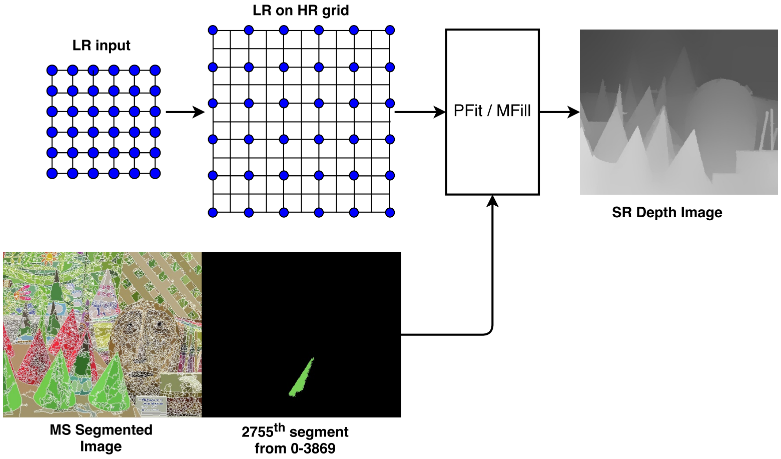


Figure 4.5: Block diagram of the proposed LRSR method using guidance colour image.

image grid such that all the points on the HR grid are at equal distance. Irrespective of the HR image resolution, the LR image points are uniformly mapped onto the HR image grid. The only difference seen in the HR image grid is the spacing between the mapped points. For upsampling factor $\times 2$, the LR image points are placed alternate on the HR image grid, but for higher upsampling factors like $\times 4$ and $\times 8$, the LR images points are mapped at every 4th or 8th point on HR image grid respectively.

Whereas, in hierarchical approach to super-resolution, the higher upsampling factor (e.g. $\times 4$, or $\times 8$) are performed in multiples of 2. Irrespective of the upsampling factor, the LR image points are laid on the HR image grid is only 2 times bigger in spatial resolution. This HR output is further mapped on the HR image grid which is 2 times bigger than itself to obtain an HR output which is totally 4 times bigger in spatial resolution. Similar chain of hierarchy is followed until the desired resolution is achieved.

As the SR upsampling factor goes higher, the results produced by hierarchical approach are better than the direct approach, because direct approach tries to estimate the unknown pixels at a larger scale, whereas hierarchical approach estimate the unknown pixels in a step of 2. The retained edges details in every steps are carry forwarded to the next iteration in hierarchical manner until the desired resolution is achieved. In this way, for higher upsampling factors, the hierarchical approach produce good results as compared to the direct approach.

The proposed method LRSR based on segment cues from HR guidance colour image using direct and hierarchical approach is shown in Figure 4.6 and Figure 4.7 respectively. In both the cases, the LR image points are laid on the HR image grid, and posed it as a depth reconstruction problem, and the earlier assumption of corresponding registered colour image holds good here too. Figure 4.6 shows direct approach to super-resolution, where, the LR image points are laid on the HR target image, and the unknown pixels are estimated by the proposed method, whereas, Figure 4.7 shows hierarchical approach to super-resolution, where, the LR image points are laid on the HR grid which is 2 times the resolution of the LR image. For higher upsampling factor, the HR output will be considered as the LR input for the next iteration, and so on. From hierarchical approach, SR for upsampling factors which are multiples of 2 can only be achieved. The pseudo code of proposed LRSR method is presented in Algorithm 3.

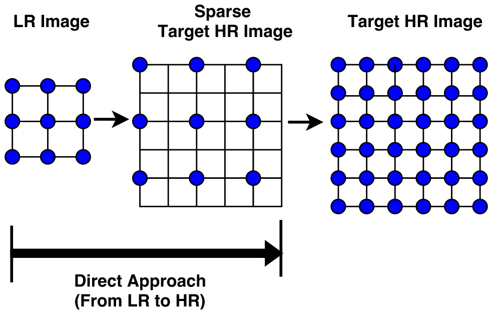


Figure 4.6: Direct approach of LRSR from LR image mapped uniformly on the HR grid.

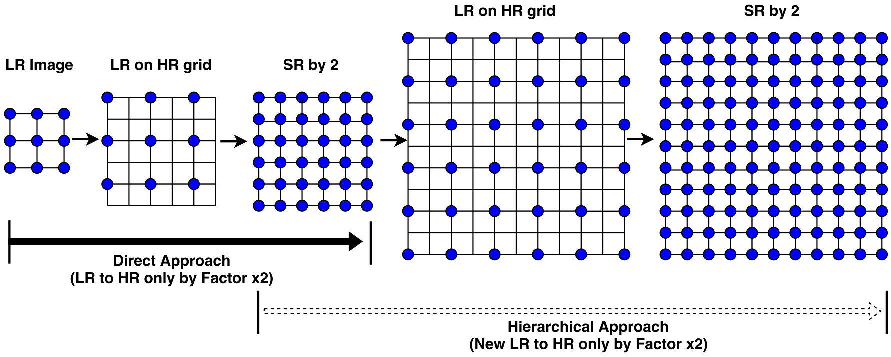


Figure 4.7: Hierarchical approach of LRSR from input LR image mapped uniformly on the HR grid of double size.

Algorithm 3 Pseudo code of proposed super-resolution (SR) from LR depth input

```
1:  Input: LR depth image $y$; Corresponding HR guidance colour image $C_x$
2:  Output: Depth SR output $\hat{x}$
3:  Ground Truth: The original HR depth image $x$
    ——— DIRECT approach ———
4:  Initialize: Set $\hat{x}$ equal to zero, with size same as $C_x$
5:  $[C_{seg}, lb]$ = segmentation($C_x$);    % MS/SLIC segmentation approach
6:  For each segment labels $lb_i$,
7:      Extract corresponding segment region in $y$, i.e. $y^{(lb_i)}$
8:      Estimate median (or plane fit) over visible pixels in the segment of $y^{(lb_i)}$, i.e.
        local_est = MFill_PFit($y^{(lb_i)}$)
9:      Fill the segment region with depth value estimated for that local segment, i.e.
        $\hat{x}^{(lb_i)}$ = local_est;
10:     Repeat steps 7-9 until all labels are addressed.
    ——— HIERARCHICAL approach ———
11: Initialize: Set $\hat{x}$ equal to zero, with twice the size of input $y$.
12: Estimate the number of hierarchical levels from SR factor (for SR by $\times 8$, 3 hierar-
    chical level for SR by $\times 2$ each)
13: For each Hierarchical level,
14:     Perform step 6-10 until target resolution is achieved.
```

4.5 EXPERIMENTAL RESULTS AND DISCUSSIONS

4.5.1 Results of LRBicSR Method

This section presents the SR results for upsampling factor $\times 4$ and $\times 8$. For experimentation purpose, some test images are chosen from popular Middlebury dataset (Scharstein and Szeliski, 2002, 2003). The SR results are demonstrated for both clean and noisy input depth images, and these results are compared with standard classical interpolation method and a popular related state-of-the-art method of guided image filtering (GIF) (He et al., 2010) which also uses RGB colour images as a guidance image. The qualitative and quantitative results are shown, and PSNR and SSIM performance metrics are used to evaluate the SR methods.

The parameters chosen in the proposed method are determined empirically by a greedy search over a range of values, and the parameters reported here are those which yielded the best quantitative results. In MS segmentation, spatial bandwidth (which decides the smoothing and connectivity of segments) and the range bandwidth (which affects the number of segments) are the crucial parameters which were set to 10 and 3 respectively, and edge strength parameter is set to 0.1. For SLIC segmentation, the only parameter to set is the number of desired superpixels k (or segments), and it is set to $k = 1500$ throughout the experiments. The threshold parameter τ, for MS algorithm, is set to 20 for noiseless images. For noisy images, the whole image is used with segment median for SR depth image reconstruction. BF filter, which has two parameters, i.e. the filter window size w and two standard deviations σ_s and σ_r for spatial and range domain filters respectively are set to $w = 7$ and (σ_s,σ_r)=(1,30). For observing the LR depth image, the LR image model from Eq. 1.3, which was discussed in Chapter 1, has been used only for LR image generation. The parameters used in the LR image model are blur filter $\mathcal{B}$), which is set to size [7$\times$7] with standard deviation 1.6, and for noisy scenario the noise η with standard deviation σ of 5 is added.

The ToF depth cameras (e.g. Kinect, Mesa swiss ranger, CanestaVision, etc.), sometimes have problems of missing pixels which will be marked as black pixels (missing pixels) in the depth image. These missing pixels were filled using the left most valid available depth value in that row. It serves the purpose of good qualitative visualization, but since these pixels are not true values they have been excluded from calculating the quantitative performances of the SR methods.

The SR results for upsampling factor $\times 4$ and $\times 8$ on *noiseless* and *noisy* images are shown below. Figure 4.8 shows the SR results on *noiseless cones* depth image, and it is seen that the overall image quality of the output produced by the proposed method looks better than the bicubic interpolation and GIF (He et al., 2010) in terms of better edge preservation. A small region from the output images is cropped and zoomed (inset in the image) to display a closer look at the details of the super-resolved images. It is seen that the bicubic interpolated results are blurred at the edges, and the GIF method produce similar kind of output with blur artifacts near edges, whereas the proposed SR method preserve edges much better than any other comparative methods. For input LR images with added noise, the SR results of which is shown in Figure 4.9 on *cones* depth image, and it is clearly seen that the noise level in the output of the proposed method is less as compared to bicubic or GIF output.

SR for higher upsampling factor is a challenge. Figure 4.10 shows SR results for upsampling factor $\times 8$ on *reindeer* depth image, and Figure 4.11 and Figure 4.12 show SR results for up by $\times 8$ on *reindeer* and *art* depth image respectively, which are *noisy*. Clearly, it is observed that the proposed method performs well in retaining the overall structure by well preserving the edges. To remove any residual noise, a BF filter employed to preserve the gained edges as well as smooth the noise. As seen in the zoomed image in last-column of Figure 4.12, the nose has sharp edge discontinuity with accurate depth and reduced noise level. Indeed, the performance of the proposed method is much better for *noisy* scenario than the *noiseless* scenario.

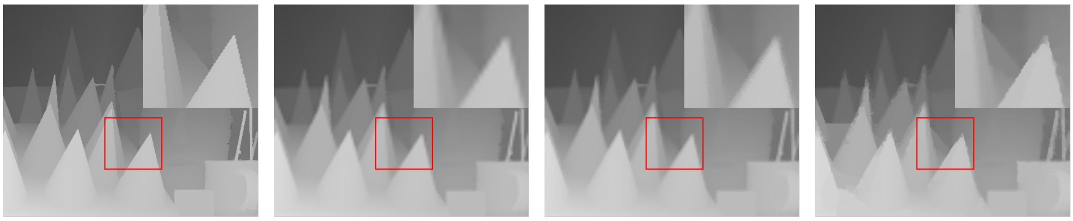

(a) GT (b) Bicubic (c) GIF (d) SR-MS-BF

Figure 4.8: SR output comparison of proposed LRBicSR method with other SR methods for upsampling factor $\times 4$ on *noiseless* image *Cones* with zoomed region inset at bottom-right corner. **From left**: GT, Bicubic, GIF, Proposed (SR-MS-BF)

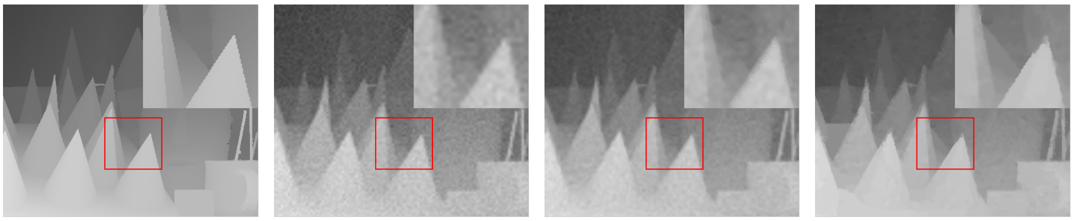
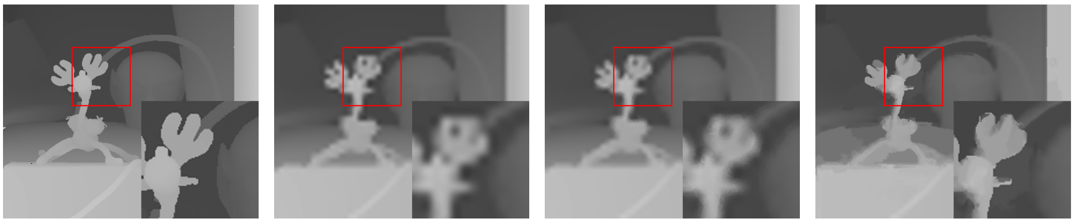

(a) GT (b) Bicubic (c) GIF (d) SR-MS-BF

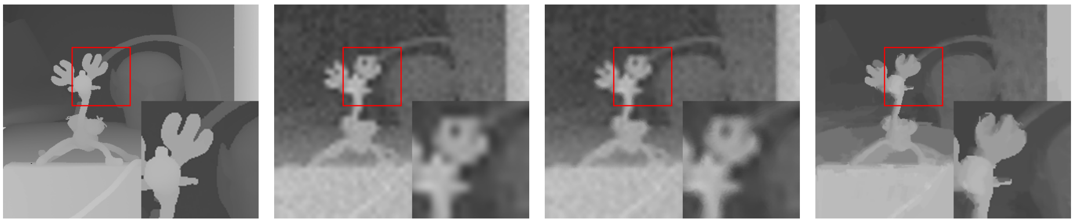
Figure 4.9: SR output comparison of proposed LRBicSR method with other SR methods for upsampling factor $\times 4$ on *noisy* image *Cones* with zoomed region inset at bottom-right corner. **From left**: GT, Bicubic, GIF, Proposed (SR-MS-BF)

To evaluate the quantitative results of the SR methods, PSNR and SSIM performance metrics are used. Table 4.1 and Table 4.2 shows the quantitative results for upsampling factor $\times 4$ and $\times 8$ respectively on noiseless and noisy images both. Notation $n0$ and $n5$ indicates noiseless and noisy (σ=5) cases respectively. The highest result in



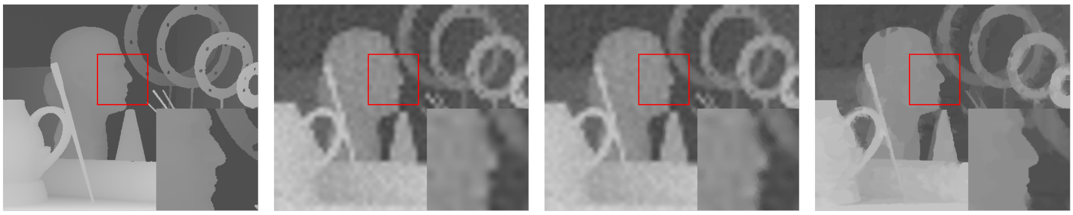
(a) GT (b) Bicubic (c) GIF (d) SR-MS-BF

Figure 4.10: SR output comparison of proposed LRBicSR method with other SR methods for upsampling factor ×8 on *noiseless* image *Reindeer* with zoomed region inset at bottom-right corner. **From left**: GT, Bicubic, GIF, Proposed (SR-MS-BF)



(a) GT (b) Bicubic (c) GIF (d) SR-MS-BF

Figure 4.11: SR output comparison of proposed LRBicSR method with other SR methods for upsampling factor ×8 on *noisy* image *Reindeer* with zoomed region inset at bottom-right corner. **From left**: GT, Bicubic, GIF, Proposed (SR-MS-BF)



(a) GT (b) Bicubic (c) GIF (d) SR-MS-BF

Figure 4.12: SR output comparison of proposed LRBicSR method with other SR methods for upsampling factor ×8 on *noisy* image *Art* with zoomed region inset at bottom-right corner. **From left**: GT, Bicubic, GIF, Proposed (SR-MS-BF)

each row for each case is marked with **bold** face. Figure 4.13 shows the comparison of average PSNR values of the proposed LRBicSR method and its variants with other SR methods.

Table show the comparison of SR results obtained from bicubic interpolation, GIF method, and some variants of proposed method. The variants of the proposed method is based on using either MS or SLIC segmentation method without or with BF filter. The proposed SR method using MS segment cues is called SR-MS, and its variant with BF filter (not as part of the proposed SR pipeline but as a post processing step) is called SR-MS-BF. Similarly, SR-SLIC and SR-SLIC-BF are variants of proposed method which uses SLIC segment cues without and with BF filter respectively. It can clearly be seen in all upsampling cases, and for noiseless and noisy scenario that the variants of proposed method involving BF filter are performing better than bicubic interpolation and GIF method. In fact, for noisy cases, all the variants (without and with BF filter) perform better. Among the variants of the proposed SR method, comparing between using MS or SLIC segment cues, MS based results are better for most of the images in both noiseless and noisy cases.

Table 4.1: Comparison of PSNR/SSIM quantitative result of the proposed LRBicSR method using MS and SLIC segment cues without and with BF filter on *noiseless* and *noisy* images with other SR methods for $\times 4$ upsampling factor

Images	Bicubic	GIF	SR-MS	SR-MS-BF	SR-SLIC	SR-SLIC-BF
	$\times$4_n0					
Cones	33.12/0.95	33.56/0.95	33.63/0.95	33.93/0.96	33.70/0.95	**34.08/0.96**
Teddy	35.78/0.97	36.26/0.97	35.94/0.97	**36.31/0.97**	35.48/0.96	35.95/0.97
Art	28.44/0.88	28.87/0.89	29.45/0.90	**29.63/0.91**	28.36/0.88	28.63/0.89
Moebius	36.53/0.96	36.89/0.96	36.54/0.96	**36.97/0.97**	35.46/0.95	36.03/0.96
Reindeer	31.22/0.95	31.70/0.95	31.30/0.94	31.49/0.95	32.07/0.95	**32.34/0.96**
Aloe	30.11/0.92	30.47/0.93	29.71/0.91	30.03/0.93	30.27/0.92	**30.61/0.93**
	$\times$4_n5					
Cones	31.39/0.86	32.44/0.91	33.01/0.92	33.55/0.95	33.12/0.93	**33.65/0.95**
Teddy	32.90/0.88	34.24/0.93	34.71/0.94	**35.47/0.96**	34.57/0.95	35.23/0.96
Art	27.72/0.80	28.36/0.85	29.19/0.89	**29.43/0.91**	28.16/0.87	28.48/0.89
Moebius	33.24/0.87	34.51/0.91	35.48/0.95	**36.23/0.96**	34.63/0.93	35.45/0.95
Reindeer	29.98/0.85	30.88/0.91	31.05/0.93	31.33/0.95	31.51/0.93	**31.91/0.95**
Aloe	29.08/0.84	29.76/0.88	29.35/0.89	29.75/0.92	29.92/0.90	**30.32/0.92**

4.5.2 Results of LRSR Method

The LRSR results for upsampling factor $\times 2$, $\times 4$ and $\times 8$ using direct and hierarchical approach using MS or SLIC segment cues are presented here. The experimental results are shown for depth images without noise ($\sigma = 0$) and with noise ($\sigma = 5$).

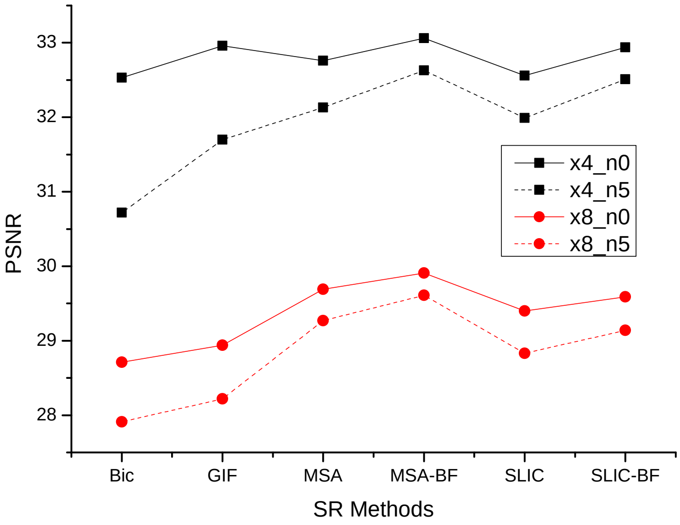
Table 4.2: Comparison of PSNR/SSIM quantitative result of the proposed LRBicSR method using MS and SLIC segment cues without and with BF filter on *noiseless* and *noisy* images with other SR methods for ×8 upsampling factor

Images	Bicubic	GIF	SR-MS	SR-MS-BF	SR-SLIC	SR-SLIC-BF
	×8_n0					
Cones	29.59/0.92	29.79/0.93	29.91/0.92	30.13/0.93	30.39/0.93	**30.60/0.93**
Teddy	31.58/0.95	31.83/0.95	32.47/0.95	**32.69/0.96**	32.13/0.95	32.31/0.96
Art	24.69/0.81	24.91/0.82	26.28/0.85	**26.44/0.87**	25.52/0.83	25.70/0.84
Moebius	32.50/0.94	32.71/0.94	33.70/0.95	**34.01/0.95**	32.68/0.93	32.95/0.94
Reindeer	27.73/0.91	28.02/0.92	29.27/0.92	**29.47/0.93**	28.83/0.92	28.98/0.93
Aloe	26.17/0.87	26.36/0.88	26.49/0.87	26.71/0.89	26.83/0.88	**27.02/0.89**
	×8_n5					
Cones	28.82/0.89	29.12/0.90	29.50/0.89	29.85/0.92	29.82/0.90	**30.15/0.92**
Teddy	30.28/0.91	30.66/0.92	31.64/0.92	**32.08/0.94**	31.20/0.92	31.53/0.94
Art	24.38/0.78	24.62/0.79	26.12/0.83	**26.30/0.86**	25.25/0.80	25.47/0.83
Moebius	31.04/0.90	31.40/0.91	33.03/0.93	**33.56/0.95**	31.85/0.90	32.33/0.93
Reindeer	27.17/0.88	27.51/0.89	29.08/0.91	**29.32/0.93**	28.32/0.89	28.56/0.91
Aloe	25.75/0.84	25.98/0.85	26.26/0.84	26.53/0.87	26.52/0.85	**26.77/0.87**

The SR results are compared among the variants of proposed LRSR method, i.e. SR-Dir-MFill, SR-Hier-MFill, both of which using either MS or SLIC segment cues, and they are also compared with bicubic interpolation method and depth map restoration from under-sampled data (SR-DRU) method (Mandal et al., 2017). SR-DRU method has used training examples of depth maps to construct a dictionary of exemplars which is used to restore the HR depth map.

The SR results of variants of the proposed SR method using MS and SLIC segment cues is compared with classical bicubic interpolation method and depth map restoration from under-sampled data (SR-DRU) method Mandal et al. (2017). SR-DRU method have used training examples of depth maps to construct a dictionary of exemplars which is used to restore the HR depth map. The SR results are shown in Figure 4.14. For clear visual comparison, their enlarged cropped region has also been shown in the next column. The images in the second column shows SR results for *noiseless* scenario, and fourth column shows SR results for *noisy* scenario. Both MS and SLIC segment cues were used for noiseless and noisy scenarios.

On noiseless images, for the SR factor of ×2, the proposed depth SR method produce results close to the bicubic interpolated images. However, the bicubic interpolation



(a)

Figure 4.13: Average PSNR result comparison of proposed LRBicSR method with other SR methods for upsampling factor ×4 and ×8 on both *noiseless* and *noisy* images.

does not consider the edges into account while super-resolving, hence the edge and corner details get blurred, unlike in the proposed method. This effect is even better visible in higher upsampling factors, e.g $\times 4$ and $\times 8$, as discussed a little later.

On noisy images, our proposed depth SR method does even better in preserving the edge discontinuities and helps in smoothing the regions with gradual depth variations, as we operate on a local segment region. Although there are some inaccuracies near edges, this is mainly because of slight bleeding of segments into the neighbouring regions which are at different depths but are similar in colour space.

Similar kind of SR results are shown in Figure 4.15 for SR upsampling factor $\times 4$. Here too, for clear visual comparison, their enlarged cropped region is shown in the next column. In this case, we compare our SR outputs using direct and hierarchical approach on noiseless and noisy images which use MS or SLIC segment cues against bicubic interpolation method and SR-DRU method Mandal et al. (2017).

As it can be seen from the Figure 4.15 that the results are more promising for upsampling factor $\times 4$ for both noiseless images (second column) and noisy images (fourth column). The second column (*noiseless* scenario) demonstrate the SR results from direct and hierarchical approach. We can observe that the object shapes and depth precision are maintained, and the edges are crisp. In the fourth column (*noisy* scenario), we can observe noise level reduction in the SR output from our proposed method. As we notice, here the edge discontinuity near the *sticks* and the *head* are sharp, and the overall object shapes and depths are maintained at its best. As noticed even earlier, the MS segment cues produce comparatively better results than using SLIC segment cues. At the end of the method, the use of bilateral filter (BF) Tomasi and Manduchi (1998) has been incorporated to smooth out any irregularities present because of dealing with local segments, but, the results after BF operation are not shown here.

Table 4.3 shows the PSNR and SSIM performance metric of SR methods on some selected test images from Middlebury dataset (Scharstein and Szeliski, 2002). The quantitative results are shown for upsampling factors $\times 2$, $\times 4$ and $\times 8$ on both noiseless and noisy images. Table 4.3 also provide results of variants of proposed method (i.e. SR-Dir, SR-Dir-BF, SR-Hier and SR-Hier-BF) using MS and SLIC segment cues which are compared among themselves and also compared with bicubic interpolation method and the SR-DRU method (Mandal et al., 2017).

As mentioned earlier, the results of proposed method are comparable to bicubic interpolation output for upsampling factor $\times 2$ on noiseless images. This is because the bicubic interpolation of depth images for a small upsampling factor (e.g. $\times 2$) does not blur the image details to a larger extent. However, as seen in all higher upsampling factors of $\times 4$ and $\times 8$ on noiseless and noisy images, the proposed method produce better results both in terms of smoothing the noise and also in terms of maintaining the edge discontinuities.

For *books* and *bowling* images, the depth SR method using MS segment cues results in a degraded output. The reason for it is that the MS segmentation approach segments the object and the background as one segment mainly because of high colour similarity in colour space; where as for the same image, our depth SR method using SLIC segment

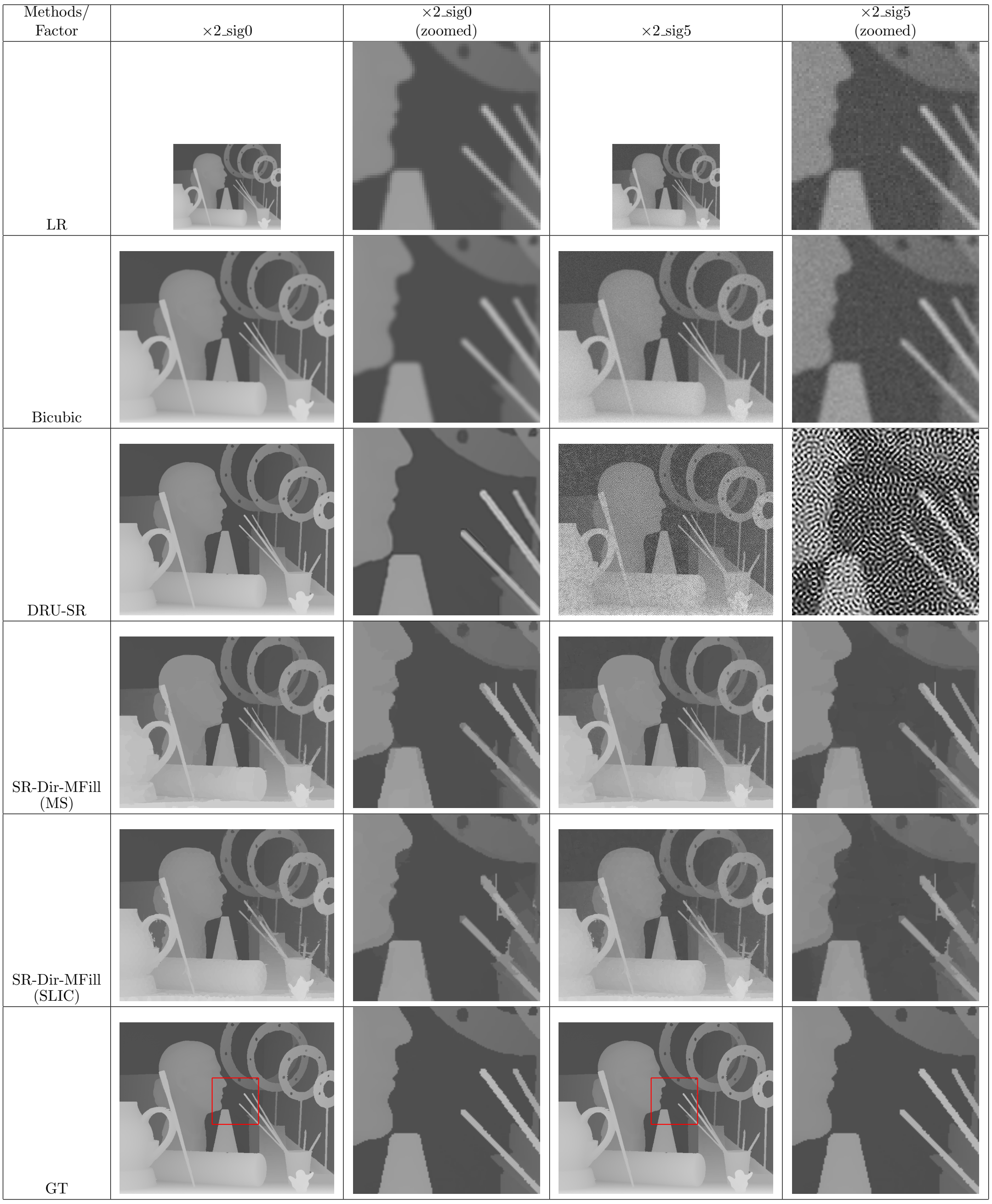


Figure 4.14: SR output comparison of proposed LRSR method and its variants with other SR methods for upsampling factor ×2 using MS and SLIC segment cues along with their cropped regions.

cues does a better job of maintaining the overall object shape and maintains the edge discontinuity, as SLIC cues performs on a super-pixels in a local segment region.

Overall, the SLIC segment cues perform better than the MS segment cues, because, the segments produced by the SLIC segmentation is finer and regular than the segments

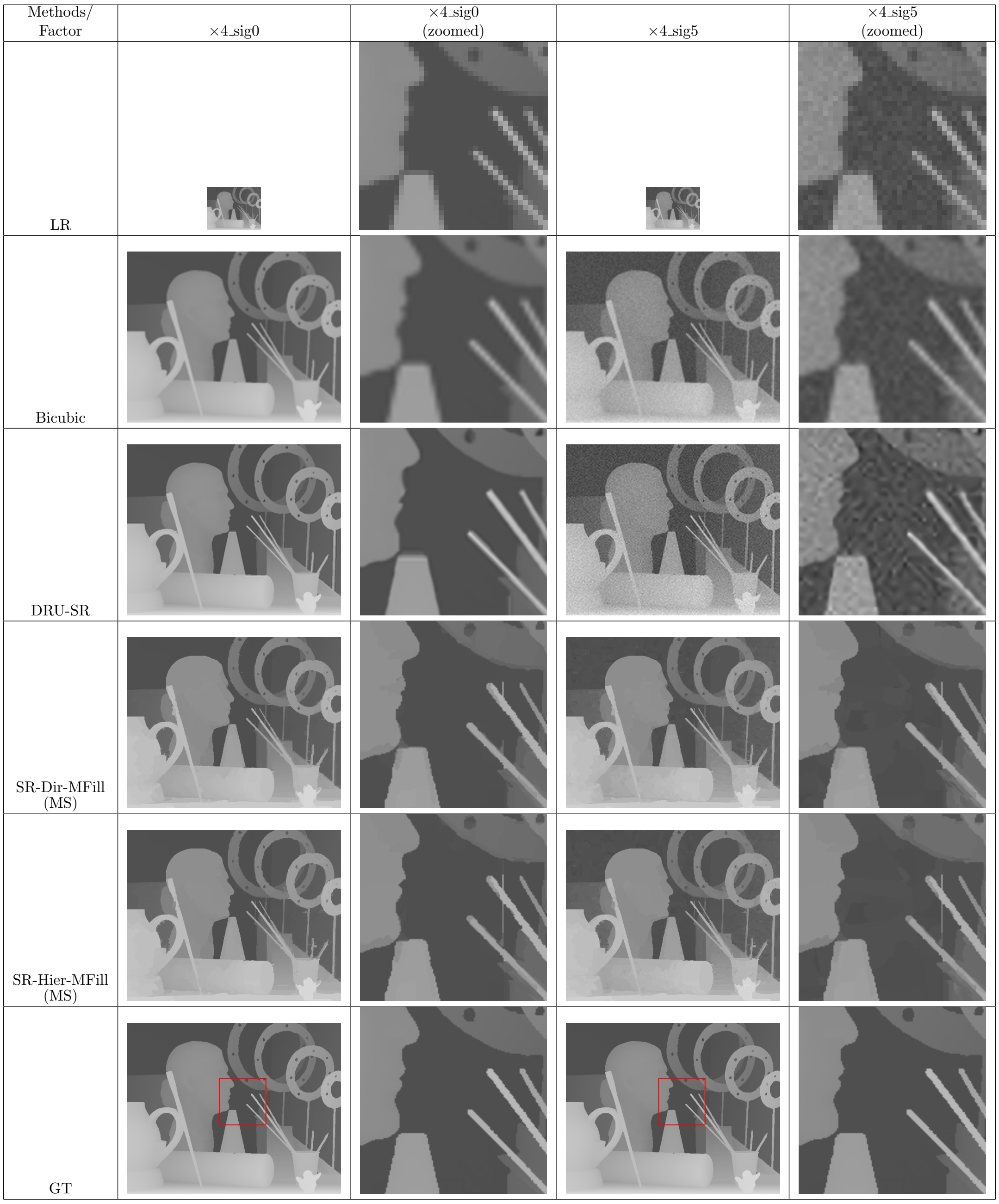


Figure 4.15: SR output comparison of proposed LRSR method and its variants with other SR methods for upsampling factor ×4 using MS segment cues by direct and hierarchical approach along with their cropped regions.

produced by the MS segmentation. Even at the high similarity colour regions between object and the background, the SLIC segmentation produces finer segment. The chances of segmenting the object region with the background region into one super-pixels are very less as compared to the number of such instances in MS segmented output. Hence, SLIC segment cue can generalize for depth images. We also note that, the MS segment cues also produces nearly similar results, but the situation get worst when the object and the background have high similarity in colour space, as opposed to the SLIC segmentation.

Table 4.3: Comparison of PSNR/SSIM quantitative result of the proposed LRSR method for upsampling factor ×2, ×4 and ×8 on depth images without and with noise. Notation $\times i\_\mathrm{sig}j$ indicates SR for upsampling factor i on images with noise standard deviation j. **First best results in bold**

SR Factor	Test Images	Bicubic	SR-DRU	MFill (MS)				MFill (SLIC)			
				SR-Dir	SR-Dir-BF	SR-Hier	SR-Hier-BF	SR-Dir	SR-Dir-BF	SR-Hier	SR-Hier-BF
x2_sig0	Aloe	35.45/0.98	**40.56/1.00**	31.68/0.96	32.31/0.96	-	-	32.07/0.96	32.90/0.97	-	-
	Art	33.27/0.98	**38.31/1.00**	32.03/0.96	32.33/0.96	-	-	31.14/0.95	31.49/0.95	-	-
	Baby	40.14/0.99	**45.05/1.00**	32.93/0.98	33.59/0.98	-	-	36.22/0.98	37.31/0.98	-	-
	Books	41.27/0.99	**45.39/1.00**	29.04/0.95	29.32/0.96	-	-	36.39/0.98	37.47/0.98	-	-
	Bowling	36.09/0.99	**42.02/1.00**	20.93/0.91	21.09/0.91	-	-	31.17/0.96	32.27/0.97	-	-
	Cones	39.47/1.00	**44.29/1.00**	36.31/0.99	36.74/0.99	-	-	37.03/0.99	37.92/0.99	-	-
	Moebius	41.85/0.99	**46.34/1.00**	36.70/0.98	37.26/0.98	-	-	37.86/0.98	38.74/0.98	-	-
	Plastic	41.33/1.00	**46.04/1.00**	30.27/0.99	30.29/0.99	-	-	40.64/0.99	41.45/0.99	-	-
	Reindeer	36.37/0.99	**41.50/1.00**	32.27/0.97	32.52/0.97	-	-	35.23/0.98	35.64/0.98	-	-
	Teddy	42.28/1.00	**46.23/1.00**	37.30/0.99	37.87/0.99	-	-	38.34/0.99	39.27/0.99	-	-
x2_sig5	Aloe	32.79/0.85	11.33/0.08	31.49/0.95	32.14/0.95	-	-	31.98/0.95	**32.76/0.96**	-	-
	Art	31.40/0.84	10.91/0.07	31.89/0.95	**32.13/0.96**	-	-	31.05/0.94	31.33/0.95	-	-
	Baby	34.62/0.83	11.07/0.04	33.06/0.97	33.76/0.98	-	-	35.98/0.97	**37.03/0.98**	-	-
	Books	34.88/0.83	10.95/0.04	29.07/0.95	29.36/0.95	-	-	36.25/0.97	**37.27/0.97**	-	-
	Bowling	**33.08/0.83**	11.15/0.06	21.02/0.91	21.17/0.91	-	-	31.20/0.96	32.24/0.96	-	-
	Cones	34.47/0.91	11.11/0.17	35.85/0.98	36.47/0.98	-	-	36.50/0.98	**37.56/0.98**	-	-
	Moebius	35.01/0.83	10.85/0.04	36.33/0.97	36.97/0.98	-	-	37.44/0.97	**38.30/0.98**	-	-
	Plastic	34.89/0.82	10.87/0.03	30.45/0.98	30.49/0.99	-	-	39.98/0.99	**40.70/0.99**	-	-
	Reindeer	33.19/0.83	10.95/0.05	32.12/0.96	32.41/0.97	-	-	34.76/0.96	**35.30/0.97**	-	-
	Teddy	35.13/0.91	10.96/0.16	36.78/0.98	37.56/0.98	-	-	37.53/0.98	**38.75/0.98**	-	-
x4_sig0	Aloe	31.84/0.96	**38.44/0.99**	31.69/0.96	32.32/0.96	30.61/0.95	31.24/0.95	32.18/0.96	33.12/0.96	30.95/0.95	31.74/0.96
	Art	30.28/0.94	**35.32/0.99**	31.79/0.96	32.20/0.96	31.27/0.95	31.84/0.96	31.30/0.95	31.73/0.96	30.50/0.95	31.00/0.95
	Baby	36.82/0.98	**42.39/1.00**	31.19/0.97	31.66/0.98	30.81/0.97	31.32/0.98	35.58/0.98	36.62/0.98	34.76/0.98	35.86/0.98
	Books	37.84/0.98	**41.55/0.99**	28.94/0.95	29.23/0.96	28.43/0.94	28.77/0.95	36.36/0.98	37.51/0.98	35.55/0.97	36.43/0.98
	Bowling	32.46/0.98	**38.88/1.00**	20.91/0.91	21.07/0.91	20.80/0.91	20.98/0.91	30.76/0.96	31.86/0.97	29.45/0.95	30.35/0.96
	Cones	36.16/0.98	**41.68/1.00**	35.97/0.99	36.47/0.99	35.48/0.98	36.15/0.98	35.71/0.98	37.48/0.98	35.40/0.98	36.21/0.98
	Moebius	38.42/0.98	**43.80/1.00**	36.53/0.98	37.09/0.98	36.08/0.98	36.73/0.98	37.93/0.98	39.19/0.98	37.26/0.97	38.30/0.98
	Plastic	37.89/0.99	**41.55/1.00**	30.35/0.99	30.38/0.99	30.25/0.99	30.28/0.99	40.34/0.99	41.27/0.99	40.25/0.99	41.20/0.99
	Reindeer	33.06/0.97	**38.46/0.99**	31.95/0.97	32.25/0.97	31.26/0.96	31.68/0.97	34.16/0.97	35.18/0.98	34.10/0.97	34.91/0.98
	Teddy	39.02/0.99	**44.13/1.00**	37.15/0.98	37.72/0.99	36.58/0.98	37.19/0.98	36.86/0.98	38.89/0.99	37.18/0.99	38.02/0.99
x4_sig5	Aloe	30.44/0.84	29.95/0.69	31.23/0.93	31.93/0.94	30.44/0.93	31.06/0.94	31.72/0.93	**32.61/0.94**	30.76/0.93	31.48/0.94
	Art	29.25/0.83	29.24/0.69	31.45/0.94	**31.85/0.95**	31.05/0.94	31.57/0.95	30.99/0.93	31.38/0.94	30.36/0.93	30.82/0.94
	Baby	33.33/0.84	30.42/0.64	31.11/0.96	31.66/0.97	30.93/0.96	31.47/0.97	34.70/0.95	**35.74/0.96**	34.18/0.96	35.21/0.97
	Books	33.82/0.84	30.40/0.65	28.94/0.94	29.26/0.95	28.16/0.94	28.52/0.95	35.56/0.96	**36.68/0.96**	34.96/0.96	35.81/0.97
	Bowling	30.93/0.84	30.16/0.65	20.98/0.91	21.14/0.91	20.84/0.91	21.01/0.91	30.59/0.94	**31.62/0.95**	29.44/0.94	30.29/0.95
	Cones	33.11/0.86	30.34/0.70	35.05/0.96	35.81/0.97	34.90/0.97	35.63/0.97	34.96/0.96	**36.67/0.97**	35.14/0.97	35.91/0.97
	Moebius	34.03/0.84	30.45/0.65	35.71/0.96	36.44/0.97	35.48/0.96	36.18/0.97	36.59/0.96	**37.74/0.96**	36.42/0.96	37.32/0.97
	Plastic	33.82/0.84	30.44/0.63	30.42/0.98	30.45/0.99	30.33/0.98	30.37/0.99	38.33/0.97	39.12/0.98	38.51/0.98	**39.31/0.98**
	Reindeer	31.26/0.84	29.92/0.66	31.69/0.95	32.04/0.96	31.12/0.96	31.54/0.96	33.48/0.94	**34.50/0.96**	33.63/0.96	34.39/0.96
	Teddy	34.24/0.86	30.49/0.68	36.01/0.97	36.87/0.97	35.95/0.97	36.64/0.98	35.86/0.96	**37.81/0.97**	36.57/0.98	37.44/0.98
x8_sig0	Aloe	27.82/0.90	31.59/0.95	30.79/0.95	31.35/0.95	29.51/0.93	30.06/0.94	31.30/0.94	32.28/0.95	30.23/0.94	30.90/0.95
	Art	26.41/0.85	28.99/0.92	30.16/0.94	**30.65/0.94**	29.63/0.93	30.11/0.94	29.91/0.93	30.47/0.94	29.18/0.92	29.55/0.93
	Baby	32.39/0.96	35.39/0.98	30.74/0.97	31.23/0.97	30.47/0.97	30.99/0.97	35.45/0.97	**36.64/0.98**	34.62/0.97	35.69/0.98
	Books	33.70/0.96	35.71/0.97	28.69/0.95	28.96/0.96	28.20/0.94	28.56/0.95	34.42/0.96	**36.13/0.97**	34.93/0.96	35.82/0.97
	Bowling	28.01/0.94	30.95/0.96	20.84/0.91	21.00/0.91	20.37/0.90	20.54/0.91	30.50/0.95	**31.50/0.96**	28.71/0.95	29.52/0.95
	Cones	31.92/0.95	34.71/0.98	35.31/0.98	**36.05/0.98**	34.32/0.97	35.09/0.97	30.80/0.94	34.40/0.96	34.14/0.98	34.99/0.98
	Moebius	34.24/0.95	37.31/0.97	35.64/0.97	36.25/0.98	35.19/0.97	35.90/0.97	35.30/0.97	**37.53/0.97**	35.58/0.97	36.42/0.97
	Plastic	33.60/0.97	35.86/0.98	30.16/0.98	30.25/0.99	30.20/0.98	30.26/0.99	37.77/0.99	38.47/0.99	38.60/0.99	**39.46/0.99**
	Reindeer	29.34/0.94	32.54/0.97	30.94/0.96	31.51/0.97	30.42/0.96	30.98/0.96	30.31/0.93	33.04/0.95	32.68/0.96	**33.50/0.97**
	Teddy	34.82/0.97	**37.87/0.99**	36.28/0.98	36.91/0.98	35.88/0.98	36.48/0.98	31.96/0.95	36.04/0.96	36.30/0.98	37.03/0.99
x8_sig5	Aloe	27.21/0.82	30.10/0.86	29.99/0.90	30.64/0.92	29.07/0.91	29.65/0.92	30.28/0.89	**31.25/0.91**	29.74/0.91	30.41/0.92
	Art	25.94/0.78	28.05/0.83	29.78/0.92	**30.30/0.92**	29.36/0.91	29.81/0.92	29.22/0.89	29.81/0.90	28.82/0.90	29.17/0.91
	Baby	30.71/0.86	32.50/0.87	30.70/0.95	31.25/0.96	30.49/0.96	31.04/0.96	33.40/0.92	**34.49/0.93**	33.31/0.94	34.26/0.95
	Books	31.62/0.87	32.66/0.86	28.63/0.94	28.94/0.95	28.24/0.93	28.60/0.94	32.92/0.91	34.38/0.93	33.77/0.94	**34.62/0.95**
	Bowling	27.35/0.85	29.72/0.86	20.91/0.91	21.07/0.91	20.53/0.90	20.70/0.91	29.81/0.90	**30.76/0.92**	28.48/0.92	29.28/0.93
	Cones	30.52/0.85	32.16/0.86	33.83/0.94	**34.68/0.95**	33.51/0.95	34.28/0.96	30.21/0.91	33.37/0.92	33.57/0.96	34.39/0.96
	Moebius	32.00/0.86	33.24/0.86	34.36/0.94	**35.08/0.95**	34.23/0.95	34.91/0.96	33.38/0.92	35.07/0.94	34.18/0.94	34.93/0.95
	Plastic	31.72/0.88	32.86/0.87	30.21/0.98	30.33/0.98	30.26/0.98	30.33/0.99	34.80/0.93	35.53/0.95	36.47/0.97	**37.24/0.97**
	Reindeer	28.46/0.85	30.78/0.86	30.58/0.94	31.18/0.95	30.36/0.94	30.94/0.95	29.55/0.88	31.95/0.91	31.99/0.94	**32.78/0.95**
	Teddy	32.34/0.86	33.53/0.86	34.74/0.95	35.53/0.95	35.09/0.96	35.73/0.97	31.16/0.91	34.52/0.93	35.10/0.96	**35.82/0.97**

4.5.3 Results of LRSR Method on Kitti Dataset

We have also performed experiments on another set of stereo image taken from Kitti dataset. We chose 5 different images from the dataset and performed same set of SR experiments on those images for two different upsampling factors $\times 2$ and $\times 4$.

Fig 4.16 shows the qualitative SR results for $\times 2$ upsampling factor for one of the image from Kitti dataset. The images shown in Fig 4.16 from top to bottom are in order as bicubic interpolated image, proposed SR image, and GT image. We can observe that the output produced by the proposed method (second row) shows much plausible image with better edge discontinuities. However, the bicubic interpolated image suffer from edge blurring. The blurness is more prominent for higher upsampling factors.

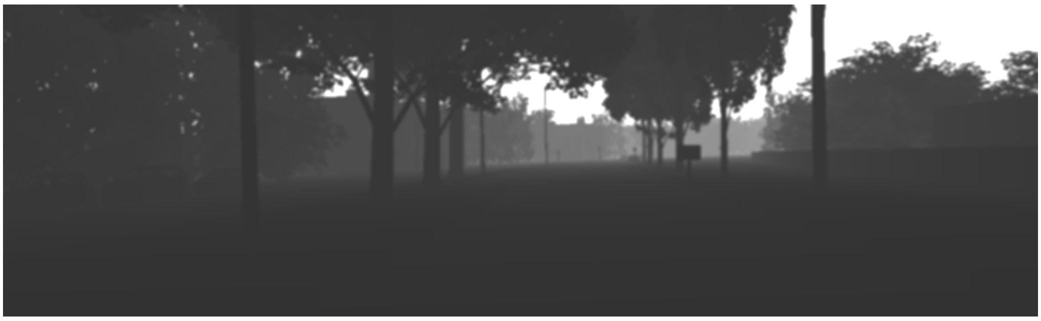
Table 4.4 shows the quantitative results of the chosen images. It shows the average PNSR and SSIM values for two different SR upsampling factors, i.e. for $\times 2$ and $\times 4$ factor. It is clearly seen that our proposed method perform well as compared to the bicubic interpolation results.

Table 4.4: Average PSNR/SSIM results of proposed LRSR method for different upsampling factors on Kitti dataset

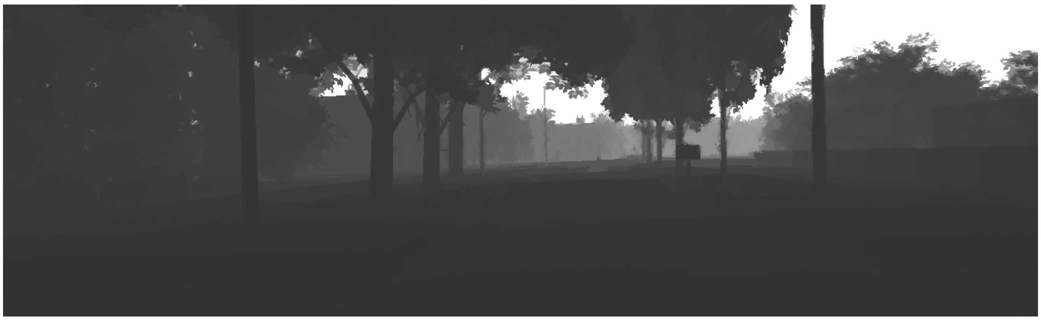
Methods ↓/Upsampling Factor →	×2	×4
Bicubic	69.28	67.47
LRSR	70.89	69.86

4.6 SUMMARY

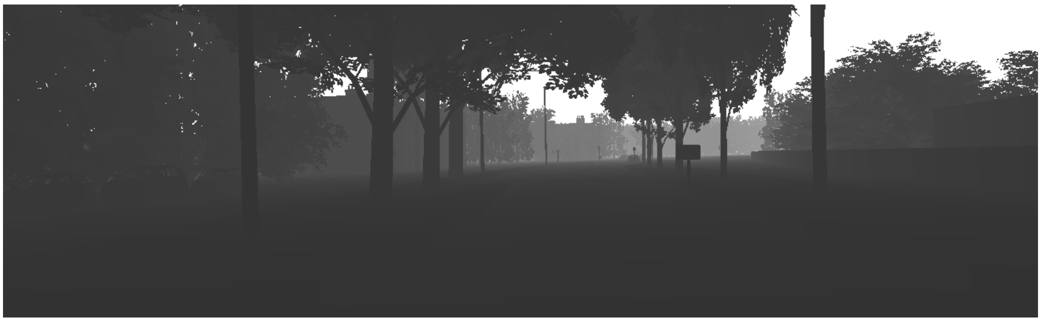
This chapter presents a HR guidance colour image based depth image super-resolution. The input for the method is an LR image, but there are two ways that this input can be utilized for different scenario for super-resolution. First way is to apply bicubic interpolation method on the LR depth input to get an estimate of the output, whereas, the second way is to map the LR depth points onto the HR image grid, which basically converts the super-resolution problem into depth reconstruction problem. There are variants of the proposed method presented in this chapter which are based on ei-



(a) Bicubic



(b) SR-Dir-MFill



(c) GT

Figure 4.16: SR output comparison of the variant of proposed LRSR method on one of the image from Kitti dataset for ×2 upsampling factor. The images shown here are adjusted for dynamic range of the depth only for the display purpose, however, the original images are darker. **From Top**: Bicubic, SR-Dir-MFill, GT.

ther using MS or SLIC segment cues, and using MFill or PFit approach for estimating unknown depth pixels, and with BF or not. For higher upsmpling factors, hierarchical approach has been shown which perform upsampling in steps of 2, which helps in getting better output with lesser artifacts at the edge discontinuities. The experimental results have been shown on both noiseless and noisy images. The proposed method performs much better for noisy images as compared to other comparative methods.

The proposed LRBicSR and LRSR methods presented here is suitable when the input LR depth image is fully observed, that means, the depth information is available at all the pixel location in an image. However, if there is a time constraint on capturing the image, then the high-end cameras are unsuitable for such task because it captures the image column-wise and it is time consuming. One can capture few random samples of the scene instead of capturing all the sample pixels, and if one could reconstruct a dense depth map from the measured samples it would save a lot of time. The next chapter is motivated by the concept of reconstructing a dense depth map from a sparsely sampled depth data from a high-end camera. It has also been shown that such an input with random sparse depth pixels can be super-resolved by cascading the DR and SR methods in a single framework. It also shows that a similar model with few changes can be used for other depth image problems like depth denoising and depth inpainting.

CHAPTER 5

RECONSTRUCTION AND SUPER-RESOLUTION OF SPARSE DEPTH IMAGE

5.1 INTRODUCTION

[1] [2] The advent of depth cameras or range scanners, have enabled acquisition of 3D measurements (or depth) of the scenes directly. However, many such high-end scanners incur high cost, and the process to scan the overall object is often time consuming. On the other hand, the low-cost devices (e.g. Kinect, time-of-flight cameras) offer real-time acquisition but yield limited resolution, noisy depth maps, and are constrained to indoor settings.

To utilize the benefits of high-end cameras and yet capture the depth images in a short time, one can capture the depth data sparsely and try to construct a dense depth image using computational approaches (Liu et al., 2015). A dense depth image reconstruction approach from less number of input samples may also be useful to improve the spatial resolution of the depth images (say from low-cost scanners), wherein the low-resolution samples can be considered as sparsely captured depth data on a high resolution grid. Such an approach can also substitute the dense stereo matching problem (given sparse depth measurements from a structure from motion pipeline).

However, for such an alternative to be truly beneficial in practice, one needs to consider the reconstruction of depth images from very less (typically random) samples (e.g. $< 10\%$) of the overall depth data. This is a challenging problem as the image grid

---

[1] Chandra Shaker Balure, Arnav Bhavsar, and M. Ramesh Kini . ”Guided Depth Image Reconstruction From Very Sparse Measurements.” *SPIE, Journal of Electronic Imaging.*

[2] Chandra Shaker Balure, Arnav Bhavsar, and M. Ramesh Kini. ”Local Segment-Based Dense Depth Reconstruction from Very Sparsely Sampled Data.” *National Conference on Communications (NCC-2017)* IEEE, 2017.

for reconstruction would have a large number of missing pixel values. In this chapter, the depth reconstruction (DR) and its super-resolution (DRSR) problem is addressed by utilizing a colour image of the corresponding scene which is registered with the depth image grid. This assumption is not impractical, as many depth cameras acquire depth image with registered colour image.

The DR problem is to find the missing pixels from the observed image with very few known pixels. The representation of DR problem is shown in Figure 5.1. The image grid with on the left side is a sparsely measured depth image and the image on the right is a densely reconstructed depth image. If the amount of visible data is as low as between 10% to 1%, it becomes more challenging to reconstruct a dense depth map even with some prior information.

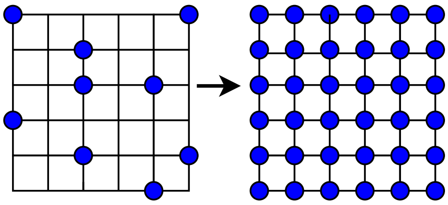


Figure 5.1: Depth reconstruction from sparse random depth pixels

To address DR problem, this chapter propose two approaches which involves only the local processing from random sparse samples. Given a registered colour image of the same scene, the proposed methods exploit a common trait of natural scenes, that is, the prominent depth edges coincide with the edges in the colour images and the depth variation within a small local region (with no depth edges) is gradual. Thus, the segmentation cues from the guidance colour image are utilized where each segment yields a local region. Ideally, these regions does not have any sudden depth changes. Such edge-based cues from colour images have indeed been employed in stereo disparity estimation approaches (Yang, 2012), and depth image super-resolution methods (Hua et al., 2016).

The first proposed method for DR problem is based on fitting a plane over the set of visible depth points within a depth segment corresponding to the segment of the colour image. All the pixels in the depth segments are filled based on local cost computations involving the segment in question and its neighboring segments, and it is represented as depth reconstruction by plane fitting (DR-PFit). As discussed later, this approach has been reported in an earlier work of (Bhavsar and Rajagopalan, 2012), but for a different scenario.

The second method for DR problem uses a median filling approach, which simply involves filling the depth segment with the estimated median values of the visible pixels in a particular segment, and here it is represented as depth reconstruction by median filling (DR-MFill). Given a very sparse depth data to start with, in both these approaches, one may face a problem of empty segments (wherein no depth data is available in a given segment), which would remain unfilled. For such problems, an iterative strategy is followed in DR-PFit, and a two-step strategy in DR-MFill to fill such empty depth segments.

Both DR-PFit and DR-MFill proposed methods have been validated on the popular Middlebury dataset (Scharstein and Szeliski, 2002) with randomized initial sampling configuration with very less visible sample data points (i.e. 10%, 5% and 1%). The DR results of proposed methods has been compared with a recent method of alternating direction method of multipliers (ADMM) (Liu et al., 2015) and depth map restoration from under-sampled data (DR-DRU) (Mandal et al., 2017). ADMM method considers exactly the same scenario of depth reconstruction from random sparse samples. Whereas, DR-DRU method is based on sparse representation by constructing sub-dictionaries from exemplar depth images. Figure 5.2 shows an example of depth reconstruction by the proposed DR-PFit and DR-MFill methods, for 1% visible random sampled depth points.

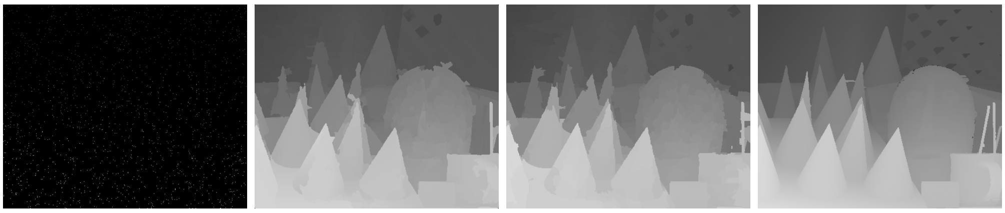

(a) 1% data (b) DR-PFit (c) DR-MFill (d) GT

Figure 5.2: Depth reconstruction from 1% random depth samples. **Left to right**: 1% sparse depth data points, PFitDR, MFillDR, GT

In this chapter, other than DR problem, the problem of super-resolution from sparse points on the LR image grid is also addressed. Such a problem is called DRSR problem, where sparse LR depth image is an input, and a densely reconstructed super-resolved image is required to produced at the output. This problem is valid in a situation where capturing the depth image takes more time and makes it undesirable for real time applications. If both dense depth reconstruction and its super-resolution can be performed from as sparse LR points, then such a solution can useful to address the real challenge of huge bandwidth requirement.

The proposed method for DRSR problem presented here is a combination of depth reconstruction (DR) and super-resolution (SR) method in a single framework. Here the DR model is cascaded to SR model to form a single model. The only assumption made here is that, the observed sparse LR image has a corresponding registered colour image. Both DR and SR modules in the DRSR framework require guidance colour image for their operation. The popular segmentation algorithms e.g. MS algorithm (Comaniciu and Meer, 2002) or SLIC algorithm (Achanta et al., 2012) is applied on the guidance colour image to obtain segment cues. Firstly, the depth reconstruction from a sparse LR image is performed, and then its output is given to the cascaded super-resolution module which maps DR module output onto the HR grid to produce SR image.

There are few other issues with depth images like noise and missing regions, and this chapter also addresses these issues by slight variation in the proposed DR framework. The proposed DR framework can be easily adapted to address these problems because the input and the output resolution is same. For depth denoising problem, the proposed DR method tries to reconstruct the depth image believing 100% visible depth pixels. Similarly, in depth inpainting problem the DR method believes that some percentage of pixels (mostly seen at the edges of the objects) are not visible because of occlusion, and it tries to reconstruct the depth image by filling the missing regions.

The work on depth image reconstruction reported by Bhavsar and Rajagopalan (2012) consider examples involving uniform as well as non-uniform sampling. However, the configuration of available depth data considered in their work is quite different than that considered here. Indeed, the plane fitting approach is a method proposed by Bhavsar and Rajagopalan (2012), but here the effectiveness is demonstrated for more challenging configurations. In the work reported by Bhavsar and Rajagopalan (2012), the uniform sampling, unlike those considered above, are in the form of regular columns or blocks of missing data, and the non-uniform sampling is similar to that in traditional image inpainting methods (e.g. irregular blocks of missing and available data). On the other hand, the configuration of available depth data in the proposed work involve depth measurements at random isolated points on the grid.

The only works, to the best of the knowledge, which considers similar randomized sampling as of the proposed work is reported by Liu et al. (2015) and Mandal et al. (2017). The proposed approach is more simplistic and efficient, and it is also been demonstrated that the proposed approach outperforms the methods presented by Liu et al. (2015) and Mandal et al. (2017). Importantly, the amount of visible data considered in other similar methods is much more than that of the proposed work. For instance, the approach of Liu et al. (2015) largely consider the amount of available data to be more than 20%-25%, whereas the maximum amount of available data in the proposed work is 10%, and it provides consistent result in examples with as low as 1% available data.

Both the proposed methods DR and DRSR fall in the category of depth reconstruction from *non-uniform* sparse samples on a uniform grid. A random sparse selection is exercised to choose the points randomly, and later use the colour segment cues for full depth reconstruction and its super-resolution.

5.2 RGB GUIDANCE IMAGE BASED DEPTH RECONSTRUCTION FROM SPARSE DATA

5.2.1 Depth Reconstruction by Plane Fitting

As mentioned earlier, depth reconstruction by plane fitting method was proposed by Bhavsar and Rajagopalan (2012) for a different (and arguably an easier) scenario of depth image reconstruction. As discussed earlier, this method existed for a different data configuration, and here it is demonstrated for a challenging configuration where the visible pixels in the input image is uncommonly as low as 1%. The block diagram of DR-PFit is shown in Figure 5.3 which is briefly discussed here for the sake of completeness.

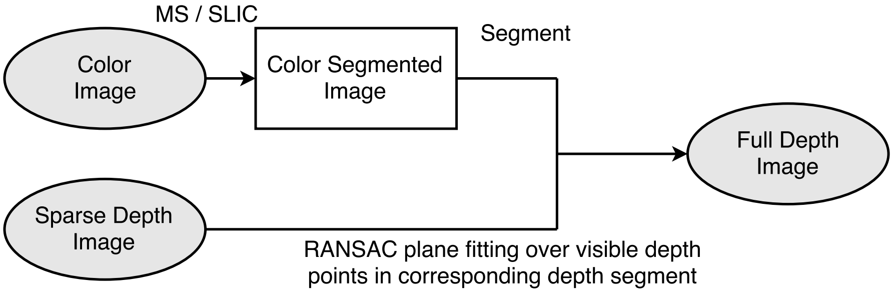


Figure 5.3: Block diagram of DR-PFit method

The DR-PFit method takes one sparse depth image and one corresponding registered colour image. A SLIC or MS segmentation method is applied on the colour image to obtain the segment cues. The segments of colour image is utilized to estimate the unknown depth values in the sparse depth image by fitting a plane over the visible pixels in that local segment region in sparse depth image. The method finds the adjacency matrix for all the segments. The adjacency matrix labels the connected segments to a particular segment. If the number of pixels in the segment are above a threshold, then plane fitting is carried out on the visible pixels using random sample consensus (RANSAC) (Fischler and Bolles, 1981) method. RANSAC is an approach which is used for robustly estimating model parameters in the presence of outliers. If the number of available pixels are below the threshold, then RANSAC will not have sufficient points for plane fitting, thus a median is computed from pixels including those from the adjacent segments.

Given a fitted plane over a segment, the local cost function for assigning a range label z for each invisible pixel p in the segment s is given in Eq. 5.1,

$$C_p = |z - z_{pl}| + \lambda_p \sum_{q \in V_p} |z - z_p| \tag{5.1}$$

where z_{pl} is the plane-fitting range at pixel location p, and V_p is the set of visible second-order neighbors of p that belong to segment s. For cases when $0 < N_v < n_{pl}$, it will have less number of pixels for plane-fitting to be robust enough. In this case, the cost function is modified as Eq. 5.2,

$$C_s = |z - z_m| + W_a \sum_{z_{m_a} \in m_a} |z - z_{m_a}| \tag{5.2}$$

where m_a is the set of medians of visible pixels of all the neighborhood segments.

DR-PFit method follows an iterative process. Firstly, it segments the colour image at a finer level, then look for the corresponding segment region in the sparse input depth image for the visible depth values. Based on the number of visible pixels, it fits a plane or find the median value to estimate the depth value of the missing pixels in that segment region. If there are still more empty segments seen at the end of the first-pass, then the segmentation algorithm segments the colour image at a coarser level, and follow the same process. The coarser segmentation combines the small empty segments in one iteration with other non-empty ones, over subsequent iterations, and thus the resultant larger segments are no more empty and can be used for plane-fitting. This continues until the depth image is completely filled. The pseudo code of DR-PFit method is shown in Algorithm 4, with sparse depth image D_s, the corresponding registered colour image C as input, and the reconstructed dense depth image D_r as output.

Algorithm 4 Pseudo code of DR-PFit method

```
Input Randomly sampled sparse depth image ($D_s$), and its corresponding RGB colour image ($C$).
Initialize: $n_{pl}$ (threshold for labeling using plane-fitting), $h_I$ (intensity bandwidth), $I_{max}$ (max. iteration)
Segment the RGB colour image using MS algo., [$imS$, $lb$] = MSSeg($C$),
for $i = 1 \cdots I_{max}$ do
    for $s = 1 \cdots n_s$ do
        Compute $N_v^s$; visible pixels in segment $s$,
        Compute $N_h^s$; hidden pixels in segment $s$,
        if $N_v^s \geq n_{pl}$ then
            Fit plane for $N_v^s$
            Label $N_h^s$ according to Eq. 5.1
        else
            Find median of $N_v^s$
            Label $N_h^s$ according to Eq. 5.2
        end if
    end for
end for
Output: Reconstructed depth image ($D_r$).
```

5.2.2 Depth Reconstruction by Median Filling

Another method proposed for depth reconstruction is by using median operation, and here this method is termed as DR-MFill. The proposed method is divided into two stages, where the first stage does a partial reconstruction, and the second stage does complete dense depth reconstruction, each of which is explained in the following text. The complete process of the proposed DR-MFill method is shown as block diagram in Figure 5.4, and the pseudo code of both the stages are shown in Algorithm 5.

The variable notations in the following explanation are described below. The variable C represents RGB colour image, D_s is randomly sampled sparse depth image, C_{seg} is the segmented image with n segments labeled as lb_i ($i = 1, \cdots, n$), e_lb_i are empty segment labels, and $idx_{circum}^{(i)}$ are indexes of boundary pixels of i^{th} segment from e_lb_i segment labels.

Stage-1: Partial reconstruction from RGB segment cue: Stage-1 starts with a courser segmentation of colour image using mean-shift (MS) algorithm (Comaniciu and Meer, 2002). For each depth region corresponding to a colour segment region, a

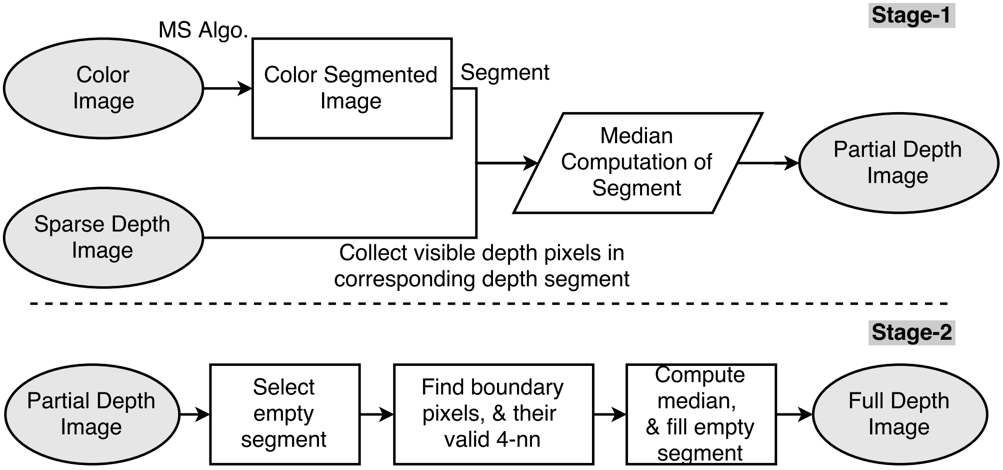

Stage-1
MS Algo.
Color Image
Color Segmented Image
Segment
Median Computation of Segment
Partial Depth Image
Sparse Depth Image
Collect visible depth pixels in corresponding depth segment
Stage-2
Partial Depth Image
Select empty segment
Find boundary pixels, & their valid 4-nn
Compute median, & fill empty segment
Full Depth Image

Figure 5.4: Block diagram of proposed DR-MFill method.

median of the visible depth value is calculated and it is assigned to all unknown depth pixels in that segment region. This approach assumes that the points a local region will have similar depth values, and thus, the median value from the available sparse points is a good approximation to estimate the missing depth pixels. This results in even simpler approach than that of the previous method of DR-PFit which is based on the locally planar assumption.

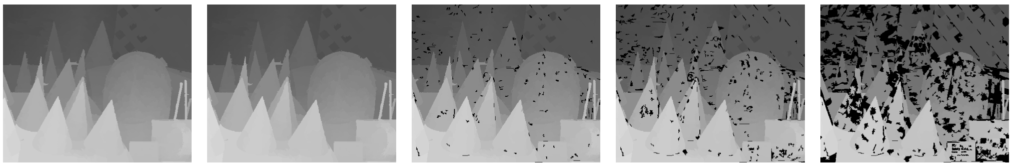
Stage-2: Empty segment filling from 4 nearest neighbors: The output of stage-1 is partial reconstructed depth output, as there can be some empty segments because of no availability of visible pixels in some segments. In such cases, the median filling does not result into approximate depth filling, but an empty region/segment.

The number of such empty segments increase with decrease in the number of samples available in the sparse depth input. Hence, the proposed method DR-MFill has another step (called stage-2) to fill these empty segments which are the residuals of stage-1. Figure 5.5 shows the reconstruction outputs for different percentage of missing data, where the first-row display results after stage-1, and the second-row display results after stage-2. It can be observed that, for 50% visible samples (first-column), there are no empty segments in the output of stage-1 because all segments generally have enough samples for reconstruction. However, as the number of samples reduces (i.e. 40%, 10%, 5%, or 1%), the stage-1 output will have more number of empty segments. These empty segments are filled using stage-2 of the proposed DR pipeline,

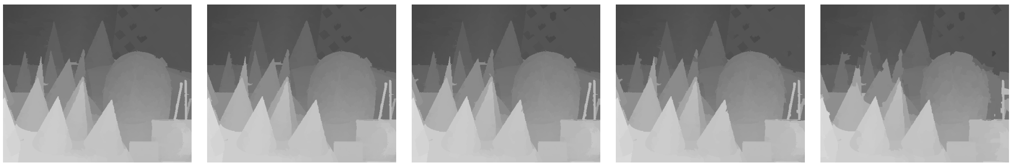
Algorithm 5 Pseudo code of DR-MFill method

```
———- STAGE-1 ———-
Input: RGB image ($C$), and randomly sampled sparse depth image ($D_s$).
[$C_{seg}$, $lb_i$] = MS_algo($C$); % Apply MS algo. on $C$
for each $lb_i, i = 1 \cdots n$ do
    $S$ = $C_{seg}(lb_i)$;        % Extract segment with label $lb_i$
    $val$ = med($D_s(lb_i)$);  % Find median of seg. $S$ in $D_s$
    $D_{partrecon}(lb_i)$ = $val$; % Fill depth seg. with $val$
end for
———- STAGE-2 ———-
Input: Depth map with empty segments ($D_{partrecon}$), and empty segment labels ($e\_lb_i, i = 1 \cdots n$)
$D_{fullrecon}$ = $D_{partrecon}$
for each $e\_lb_i, i = 1 \cdots n$ do
    $idx^{(i)}_{circum}$ = empty_seg_bd($D_{partrecon}(e\_lb_i)$);
    $vec$ = nn4($idx^{(i)}_{circum}$);  % Find 4-nn of each bd pixel
    $val$ = median($vec$);   % Find median
    $D_{fullrecon}(lb_i)$ = $val$;  % Fill segment with median
end for
```

and the PSNR values of each of the reconstructed image is shown below the individual images. It is well understood that, as the sparseness in the input depth image increases, the reconstruction accuracy also decreases, which can be clearly seen from the PSNR values of the reconstructed image shown after stage-2.



(a) 50% samples (b) 40% samples (c) 10% samples (d) 5% samples (e) 1% samples



(f) 34.14/0.96 (g) 34.12/0.96 (h) 34.01/0.96 (i) 33.46/0.95 (j) 30.51/0.94

Figure 5.5: Output of proposed two-stage DR-MFill method for decreasing samples (L to R: 50%, 40%, 10%, 5% and 1% visible pixel). **Top row**: stage-1 output, and the effect on the number of empty segments. **Bottom row**: stage-2 output, and their corresponding PSNR and SSIM values.

Stage-2 tries to inpaint the empty segments in the output image of stage-1. For each empty segment, it finds its boundary pixel locations ($idx^{(i)}_{circum}$), where the superscript i represent the segment (label) number. It then look for valid (visible) 4 nearest neighbors of the boundary pixel location, which is exercised for all the boundary pixels. A median is calculated over all such valid neighboring pixels, and then fill in the empty segment with the calculated median value. As these empty segment are largely observed in the smooth regions, so the neighborhood values are quite similar to each other. Moreover, the size of such empty segments is also typically small, hence a constant depth assumption is valid for these as well. Thus, a median estimation is a simple and an effective choice to be used to fill the empty segments. The effectiveness of stage-2 for filling in the empty regions, can be seen in second-row of Figure 5.5, where there are very few artifacts seen even in case of many missing segments in output of stage-1.

5.3 RGB GUIDANCE IMAGE BASED DEPTH IMAGE SR FROM SPARSE LR INPUT

The proposed DRSR method is a cascade of DR and SR methods in a single framework. Figure 5.6 depicts the proposed *cascade* DRSR approach for dense reconstruction and and its super-resolution from sparse LR depth image. Input to DRSR method is a sparse LR depth image (D_{LRPC}) and its corresponding registered colour image (C_{LR}). The DR module in the DRSR framework produces a dense depth map (called D_{LR}) using segment cues obtained from its guidance colour image. The output D_{LR} of DR module is considered as input to the SR module where it first maps the LR pixels on the HR image grid and then estimate the unknown pixels using median filling approach which was explained earlier.

The crux of the proposed DRSR method is that, in DR module, for each local segment region of the colour image C_{LR}, a corresponding segment region in D_{LRPC} is looked to estimate the unknown depth values using the earlier proposed median filling (MFill) approach. The output image D_{LR} of DR method is considered as the input to the cascaded SR method in a DRSR framework. The LR image D_{LR} is mapped on

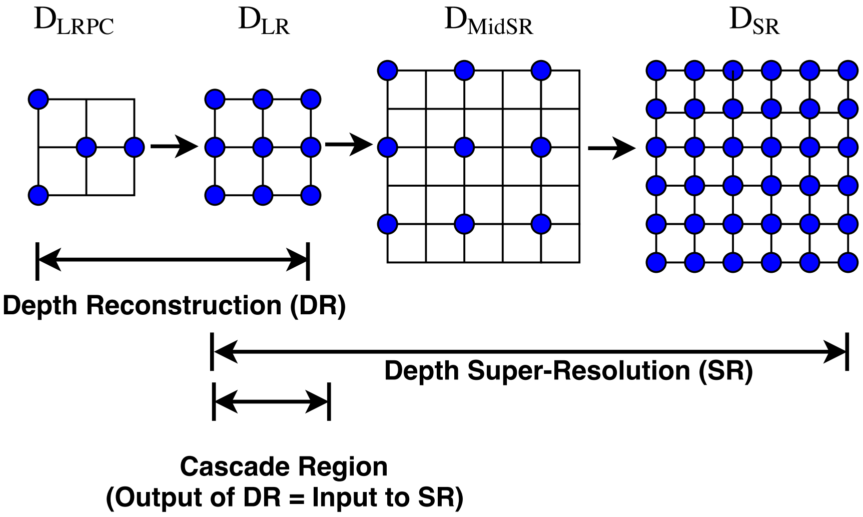

D_{LRPC}
D_{LR}
D_{MidSR}
D_{SR}
Depth Reconstruction (DR)
Depth Super-Resolution (SR)
Cascade Region
(Output of DR = Input to SR)

Figure 5.6: Super-resolution from sparse depth points

the HR grid D_{MidSR} of required size. Here also, like earlier, the corresponding colour images local segments are used as cue to estimate the unknowns on the HR grid. For higher SR upsampling factors, a hierarchical approach has been presented as direct approach produce blurry artifacts because of the reason that it need to estimate the values of many unknown depth points between the known points.

The complete pseudo code of the proposed DRSR method is shown in Algorithm 6, where y is the sparse LR input image which is the first input, and C_x is the HR guidance colour image which is the second input. The DR module takes the input y and its corresponding colour image C_y of same size as that of y (i.e. C_y is downsampled version of C_x to the required resolution), and it produces a densely reconstructed depth image $\hat{y}$. The DR output $\hat{y}$ is then fed as an input to the SR module which produces a super-resolved output $\hat{x}$ which is q times the input resolution in both x- and y-direction. The entire algorithm shown in Algorithm 6 is split into two parts, one is the DR module and other one is the SR module.

Algorithm 6 Pseudo code of proposed depth reconstruction and its super-resolution (DRSR) from sparse LR depth input

```
 1: Input Sparse LR image $y$; Corresponding HR guidance colour image $C_x$
 2: Output: Densely reconstruction super-resolved output $\hat{x}$
 3: Ground Truth: The original HR depth image $x$
    ———- Depth Reconstruction (DR) Module ———-
 4: Initialize: Set $\hat{y}$ as dummy output with size equal to $y$.
 5: $[C_{seg}, lb]$ = segmentation($C_y$);    % MS/SLIC segmentation approach
 6: For each segment labels $lb_i$
 7:     Extract corresponding segment region in $y$, i.e. $y^{(lb_i)}$;
 8:     Estimate median (or plane fit) over visible pixels in segment $y^{(lb_i)}$, i.e. local_est
        = MFill_PFit($y^{(lb_i)}$)
 9:     Fill the segment with estimated local depth value, $\hat{y}^{(lb_i)}$ = local_est;
10:     Repeat steps 7-9 until all labels are addressed.
    ———- Depth Super-Resolution (SR) Module ———-
    DIRECT approach
11: Initialize: Initialize $\hat{x}$ with $\hat{y}$, such that depth values of $\hat{y}$ are spread uniformly on
    target HR image grid $\hat{x}$.
12: $\hat{y}$ is laid on the HR grid to produce intermediate output, $\hat{x}_{mid}$
13: $[C_{seg}, lb]$ = segmentation($C_x$);    % MS/SLIC segmentation approach
14: For each segment labels $lb_i$,
15:     Extract corresponding segment regions in $\hat{x}$
16:     Estimate the median (or plane fit) over visible pixels in that local segment $\hat{x}^{(lb_i)}$
17:     Fill the segment region with estimated depth values.
18:     Repeat step 15-17 until all labels are addressed.
    HIERARCHICAL approach
19: Initialize: Initialize $\hat{x}$ with $\hat{y}$, such that $\hat{x}$ is twice the size of $\hat{y}$ with pixels uniformly
    spread.
20: Estimate the number of hierarchical levels from SR factor (for SR by ×8, 3 hierarchical levels)
21: For each Hierarchical level,
22:     Perform step 14-18 until target resolution is achieved.
```

5.4 RELEVANT APPLICATIONS BASED ON PROPOSED APPROACH

There are few other problems with depth images i.e. noise and missing regions. The noise in the image gets added during image capturing. There are many parameters which are internal and external to camera which are responsible for noise in an image. The mission regions is another issue where there won't be any depth information about the scene because of the scene occlusion. There are several methods available in the literature on depth denoising and depth inpainting.

Here, the problem of depth image denoising and inpainting is considered as the problem of depth reconstruction. In this section we show that how the proposed guidance based method using segment cue can also be used to address the problem of depth denoising and depth inpainting.

5.4.1 Depth Denoising

Depth denoising is a critical problem. For denoising the depth image we have estimate the noisy pixels and replace it with non-noisy (or nearly true) pixel values. It is shown here that how the proposed guidance based method can be easily adapted for the task of depth image denoising.

As discussed earlier, the guidance based method takes two inputs, and here also the first input is a noisy depth image and the second image is its corresponding color image as guidance image. The guidance colour image is segmented using MS or SLIC segmentation approach, in a similar way as it was done in DR and SR methods mentioned earlier. For each of the segment region obtained from the guidance colour image, a corresponding segment region in noisy depth image is looked. Now, using all the pixels in that local segment region, a median value is estimated to fill the whole segment region with the estimated value.

The above mentioned approach is essentially a local (segment-level) median filtering method, which respects the depth discontinuities, which are preserved (unlike the traditional filtering method), due to the use of the segmentation cue. Similar to using the median estimate over the segment, one can also use a plane-fit estimate. However, it is observed that the plane fitting approach would consider noisy pixels for fitting the plane over the visible pixels, and hence may be error-prone. In this work, a normally distributed noise is considered.

5.4.2 Depth Inpainting

Another common concern with depth images from depth cameras is that they suffer from having missing regions. The main reasons for such depleted regions are occlusion and poor surface reflection. The camera is not able to view a part of the scene because the reflected rays from the object are not able to reach it due to occlusion or poor or non-uniform reflectivity. To address the task of filling such missing regions with plausible information, there are many inpainting methods reported in literature Liu et al. (2012); Qi et al. (2013); Herrera et al. (2013); Bhattacharya et al. (2014).

As in the denoising case, here too, the proposed approach can be applied to address the depth inpainting problem. The approach for inpainting is similar to the proposed DR method, but unlike DR, the inpainting problem has only one variation that it looks only for those segments which has at least a single missing pixel which represents missing depth. All other segment regions, where there are no missing pixels, are left untouched. For those segments with missing depth pixels, either median filling or plane fitting approach over the visible pixels in that local segment region is calculated to estimate the missing pixels. However, unlike DR problem (where the visible pixels are distributed randomly), in the case of inpainting the size of the contiguous missing region can be larger. Thus, the order of considering the segments for operation is important. The partially filled segments are inpainted first, which are mostly found at the edge of the missing regions, and lated move towards the interior as more and more segments get filled.

5.5 EXPERIMENTAL RESULTS AND DISCUSSIONS

5.5.1 Results of DR

This section demonstrate the results of proposed depth reconstruction methods DR-PFit and DR-MFill. The depth reconstruction results of DR-PFit and DR-MFill method using MS and SLIC segmentation method for generating segment cues for dense depth reconstruction from 10%, 5% and 1% random visible depth pixels are shown in Figure 5.7, Figure 5.8 and Figure 5.9 respectively. The proposed method has been validated on some selected depth images from popular Middlebury dataset (Scharstein and Szeliski, 2002), which consists of depth images and its corresponding registered colour images, with variety of variations in the images. The proposed method has been compared with state-of-the-art depth reconstruction methods like alternating direction method of multipliers (ADMM) method (Liu et al., 2015) and depth map restoration from under-sampled data (DR-DRU) method (Mandal et al., 2017). The ADMM method use a set of depth training examples for learning the wavelet and contourlet coefficients for dense depth reconstruction, whereas the DR-DRU method is based on sparse representation by constructing sub-dictionaries from exemplar depth images. The available source code of ADMM and DR-DRU has been used for obtaining their results. Other than state-of-the-art methods, the comparison is also shown among the variants of the proposed method which are based on the approach of estimating the unknown depth values (i.e. PFit or MFill), and also based on the segmentation method used for segmenting the guidance colour image (i.e. MS or SLIC). The variants of proposed method are named as DR-PFit-MS (which use plane fitting approach with MS segment cues), DR-MFill-MS (which use median filling approach with MS segment cues), DR-PFit-SLIC (which use plane fitting approach with SLIC segment cues) and DR-MFill-SLIC (which use median filling approach with SLIC segment cues). The comparative results are shown both qualitatively and quantitatively.

Figure 5.7 shows the results of depth reconstruction from 10% observed depth pixels and the remaining 90% depth pixels are missing which needs to be estimated. The images shown in first-row are the sparse depth input, ground truth image, and depth reconstructed output from ADMM and and DR-DRU methods respectively. The images in the second-row are the depth reconstructed output obtained from the variants of the proposed method. It is observed that the images in the second-row have better edge discontinuities. ADMM method shows missing regions (e.g. *sticks on the right side*) and DR-DRU method is unable to preserve the edge discontinuities, instead it shows jaggedness at the edges.

As the sparsity of the image increases (i.e. lesser visible pixels), the reconstruction becomes more challenging. Figure 5.8 shows the DR results from 5% observed sparse depth pixels. It can be shown that the reconstructed image produced by the variants of the proposed method shown in second-row of Figure 5.8 are still able to perform better than the ADMM and DR-DRU methods. The outputs of ADMM and DR-DRU suffer heavily at the edge discontinuities which is unacceptable from any depth reconstruction method.

Figure 5.9 show even more challenging situation of depth reconstruction from just 1% observed sparse depth image where 99% of the depth pixels are to be estimated. Figure 5.9, it is clearly seen that the variants of the proposed method shown in second-row outperform ADMM and DR-DRU method. The ADMM method is not even able to preserve the object shape. The output shows artifacts similar to fast moving object with trailing effect. The DR-DRU method is also not able to preserve the edges of the object and it has lot of ghost like artifacts. On the other hand, both the proposed methods DR-PFit and DR-MFill approaches does well in terms of retaining the overall shape of the object and has a consistent depth variation throughout the image. However, there are some missing regions seen in the output even after two stage of DR-MFill approach. The reason is that, as the percentage of observed pixels in an image goes very low e.g. 1% of depth data, the SLIC segments region correspondence in the depth image might not have any valid depths pixels in its 4-neighborhoods, thus the median estimation will have no valid value in that segment. However, the proposed approach with MS segmentation is able to successfully reconstruct the depth images reasonably well even with very high sparsity (1% visible pixels) in the input depth image.

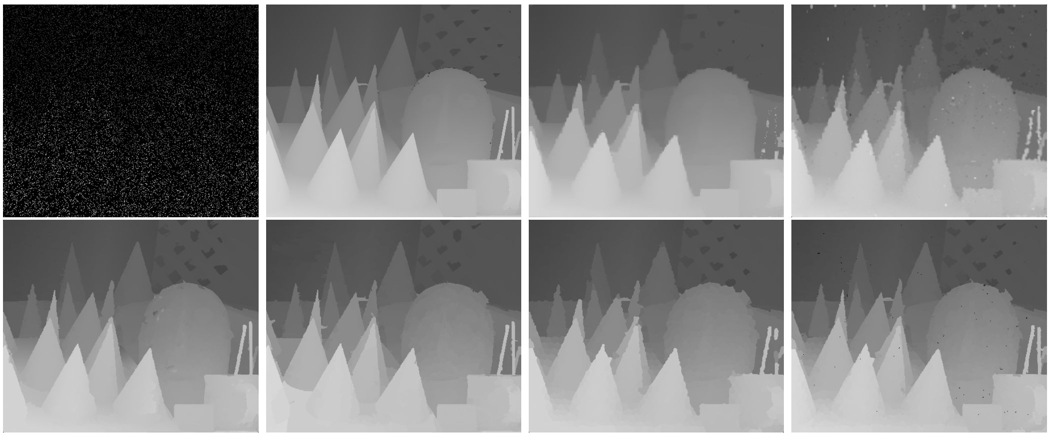


Figure 5.7: Depth reconstruction results of image *cones* from only 10% visible random depth pixels. **Top-row**: Sparse depth input, GT, ADMM, DR-DRU; **Bottom-row**: DR-PFit-MS, DR-MFill-MS, DR-PFit-SLIC, DR-MFill-SLIC

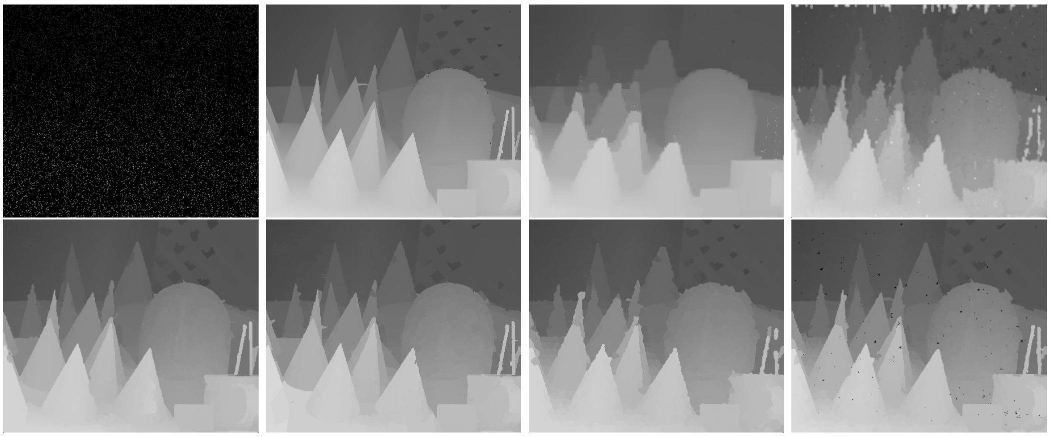


Figure 5.8: Depth reconstruction results of image *cones* from only 5% visible random depth pixels. **Top-row**: Sparse depth input, GT, ADMM, DR-DRU; **Bottom-row**: DR-PFit-MS, DR-MFill-MS, DR-PFit-SLIC, DR-MFill-SLIC

Table 5.1 shows quantitative results in terms of PSNR and SSIM metrics for 10%, 5% and 1% sparse depth image. The 1% scenario is a challenging task of reconstruction, because it has very less visible depth samples. Table 5.1 shows the comparison of depth reconstruction methods ADMM, DR-DRU and variants of proposed methods like DR-PFit-MS, DR-MFill-MS, DR-PFit-SLIC and DR-MFill-SLIC. The **bold** text in the table show the highest PSNR among different depth reconstruction methods. ADMM method shows good performance with 50% sparse depth image (not shown in Table 5.1), but the reconstruction performance decreases with decrease in the percentage of observed

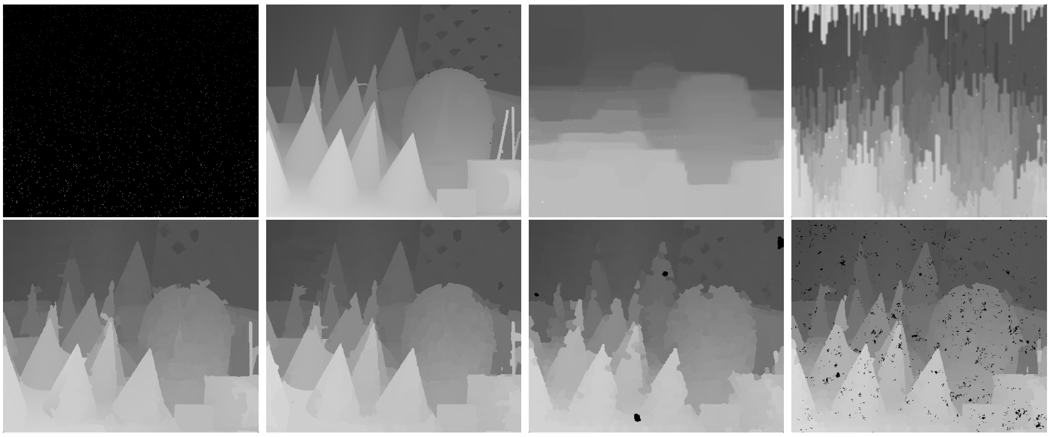


Figure 5.9: Depth reconstruction results of image *cones* from only 1% visible random depth pixels. **Top-row**: Sparse depth input, GT, ADMM, DR-DRU; **Bottom-row**: DR-PFit-MS, DR-MFill-MS, DR-PFit-SLIC, DR-MFill-SLIC

visible pixels in the sparse depth image. The first set of results shown in Table 5.1 are from 10% sparse depth pixels. It is observe that one of the variants of the proposed method (DR-PFit-MS) performs equally well as compared to ADMM method. However, as the percentage of available pixels lowers to 5% and 1%, the results of ADMM and DR-DRU decline in quality and the variants of proposed method surpasses them. For majority of the test images, the DR method using the MS segmentation (i.e. DR-MFill-MS or DR-PFit-MS) works better than the DR method using SLIC segmentation (i.e. DR-MFill-SLIC or DR-PFit-SLIC), because the MS segmentation produce better edge aligned segments as compared to SLIC segmentation. On the other hand, MS segment cues can be troublesome if the segments are of large region which might disobey the depth precision (e.g. floor or wall ceiling).

As shown in Table 5.1, for 10% sparse depth images, the DR-PFit-MS method perform better than the ADMM approach for few test images, and perform better than DR-DRU for all test images. However, as the number of observed samples in an image goes lower (i.e. 5% or 1%), the plane fitting and median filling approaches does a much better job of depth reconstruction as compared to ADMM and DR-DRU. While DR-PFit-MS performs the better for most of the test images in 5% scenario, even a much simpler method of DR-MFill-SLIC performs better than a more sophisticated ADMM and DR-DRU method.

Depth reconstruction from 1% data is even a greater challenge, and variants of proposed method consistently yields superior results over ADMM and DR-DRU methods. Indeed, for 1% sampling case, it is observed that the median filling approach (i,e DR-MFill-MS or DR-MFill-SLIC) perform better than the plane fitting approach (i.e. DR-PFit-MS or DR-MFill-SLIC). This could be because of unreliability of plane-fitting for few segments in the case of very less available data. Overall, it is seen that both plane fitting and median filling methods perform better as the number of samples go lower.

Table 5.1: PSNR/SSIM results of depth reconstruction (DR) from 10%, 5% and 1% visible data using MS and SLIC segment cues. Best results are in **bold**

Test Images	ADMM	DR-DRU	DR-PFit (MS)	DR-MFill (MS)	DR-PFit (SLIC)	DR-MFill (SLIC)
	10% visible data					
Aloe	**31.11/0.95**	30.37/0.92	24.94/0.91	29.21/0.92	21.61/0.80	30.26/0.93
Art	28.37/0.91	28.65/0.89	**30.39/0.93**	29.68/0.92	27.80/0.89	29.31/0.91
Baby	**37.44/0.98**	34.82/0.95	31.17/0.97	31.27/0.96	32.36/0.96	34.76/0.97
Books	**40.67/0.99**	34.25/0.96	33.57/0.96	29.72/0.95	34.88/0.96	35.49/0.96
Cones	32.84/0.96	33.15/0.94	**35.67/0.97**	34.01/0.96	32.80/0.95	33.24/0.95
Moebius	**38.30/0.97**	34.39/0.95	35.86/0.96	35.10/0.95	35.36/0.95	36.71/0.96
Plastic	**44.64/0.99**	34.82/0.93	37.62/0.99	30.17/0.98	39.01/0.98	42.93/0.99
Reindeer	32.28/0.96	31.32/0.94	**33.90/0.97**	30.89/0.95	31.90/0.95	33.42/0.95
Sawtooth	36.50/0.98	34.02/0.95	20.52/0.90	**36.62/0.98**	21.47/0.77	35.62/0.97
Teddy	37.09/0.98	33.57/0.93	**37.44/0.98**	34.91/0.96	34.51/0.96	33.15/0.95
	5% visible data					
Aloe	27.78/0.92	27.52/0.89	25.49/0.89	28.99/0.92	21.86/0.81	**29.73/0.92**
Art	25.89/0.87	25.53/0.84	**29.73/0.92**	29.27/0.91	27.48/0.88	28.70/0.90
Baby	34.10/0.97	32.31/0.94	30.60/0.97	31.06/0.96	31.89/0.95	**34.42/0.96**
Books	**37.57/0.98**	28.77/0.93	32.19/0.95	29.65/0.95	33.86/0.95	34.72/0.95
Cones	30.74/0.95	29.22/0.92	**34.53/0.96**	33.46/0.95	32.28/0.94	30.58/0.92
Moebius	34.28/0.96	29.11/0.92	**36.29/0.96**	34.94/0.95	31.67/0.94	35.00/0.95
Plastic	41.57/0.99	30.54/0.92	38.10/0.99	30.15/0.98	37.86/0.98	**41.89/0.99**
Reindeer	29.51/0.95	27.63/0.91	**32.43/0.96**	30.53/0.95	31.39/0.94	32.23/0.94
Sawtooth	33.53/0.97	30.55/0.94	20.48/0.90	**36.26/0.98**	21.51/0.77	33.30/0.95
Teddy	33.96/0.96	28.47/0.91	**36.56/0.97**	34.88/0.96	33.57/0.95	29.99/0.91
	1% visible data					
Aloe	21.34/0.85	21.43/0.74	23.91/0.85	**27.27/0.90**	20.55/0.78	26.74/0.84
Art	20.08/0.78	18.18/0.63	25.36/0.87	**26.95/0.88**	23.41/0.82	24.21/0.84
Baby	27.15/0.94	24.56/0.86	29.41/0.96	30.12/0.96	26.40/0.91	**29.80/0.92**
Books	28.02/0.95	19.69/0.78	30.42/0.94	29.00/0.94	25.06/0.93	**30.92/0.93**
Cones	25.70/0.91	19.89/0.77	**31.18/0.94**	30.51/0.94	27.91/0.92	22.43/0.75
Moebius	22.67/0.91	20.13/0.78	**32.10/0.94**	32.07/0.94	27.80/0.90	26.84/0.91
Plastic	30.57/0.97	22.18/0.85	35.25/0.98	29.77/0.98	28.77/0.95	**37.72/0.98**
Reindeer	23.49/0.91	20.29/0.76	26.48/0.94	**29.00/0.94**	26.96/0.90	25.15/0.84
Sawtooth	25.69/0.93	22.12/0.81	21.12/0.90	**35.06/0.97**	20.76/0.74	26.19/0.79
Teddy	25.87/0.93	21.11/0.78	**32.94/0.95**	32.72/0.95	28.95/0.93	22.60/0.79

5.5.2 Results of DRSR

Parameter Selection:

For guidance colour image segmentation, MS or SLIC segmentation methods have been used which involve some parameters setting. For MS algorithm, there are two specific parameters, i.e. spatial bandwidth (s) and range bandwidth (r). The spatial bandwidth affects the smoothing and the segments connectivity, and the range bandwidth affects the number of segments. For SLIC segmentation method, the parameters are the number of super-pixels (k) and the weighting factor between the colour and spatial differences (m).

The above mentioned parameters for MS and SLIC methods are set empirically to suit the need for the tasks at hand, and it is kept constant throughout the experiment of that particular task. The parameter values were chosen using greedy method where several values were tried and the value which gave best result was chosen. The proposed SR and DRSR methods have also been implemented using iterative approach as well as direct approach, and for each iteration the segmentation parameters were changed as per the upsampling factor and the iteration count, which again were set empirically and kept constant throughout the experiment of that particular case. The empirically chosen parameters are shown in Table 5.2.

Table 5.2: Parameters used in the proposed method

Parameters	DR method	SR method			DRSR method			Denoising	Inpainting	
		for ×2	for ×4 (Hier)	for ×8 (Hier)	for ×2	for ×4 (Hier)	for ×8 (Hier)		M.bury	Scratch
MS spatial BW	7	7	6,7	5,6,7	7	6,7	5,6,7	10	7	7
MS range BW	6.5	5	4,5	3,4,5	5	4,5	3,4,5	3	3	3
MS PFit thresh	50	50	50	50	50	50	50	50	50	50
SLIC sp	1500	2000	1800,2000	1600,1800,2000	6000	4000,6000	2000,4000,6000	1000	4000	2000
SLIC wt factor	5	5	4,5	3,4,5	5	4,5	3,4,5	5	5	10
SLIC PFit thresh	50	50	50	50	50	50	50	50	50	50

In the following section, the DRSR results are presented and compared with other state-of-the-art methods. Both qualitative and quantitative results are shown.

Middlebury dataset (Scharstein and Szeliski, 2002) contains three classes of images, and each class has some set of images but with different spatial resolution. These class of images have *one-third* ($\sim$ 450 × 350), *one-half* ($\sim$ 650 × 550), and *full* ($\sim$ 1300 × 1100) resolution images. For all the work mentioned before, we have used *one-third* sized images, but for DRSR we have used *one-half* sized images. Had we used *one-third* sized images, the DRSR method would suffer for higher upsampling factor (e.g. $\times 8$). Because, the LR image generation from the GT image in *one-third* sized image dataset would be approximately 55×45. For that much small size of LR image, the guidance colour image also need to be downsampled, because in DRSR pipeline, the DR stage require depth input and the guidance colour image to be of same spatial resolution. Now, if MS/SLIC segmentation is applied to the guidance colour image of such a small spatial resolution, the objects segmentation in the colour image will be very coarser, hence the dense depth reconstruction will produce output with lesser details in the DR output. Now, if such a degraded DR output is given to the cascaded SR stage of DRSR pipeline, the output image will be even more degraded. Hence, *one-half* sized images were used for the experiments, and the shown results for DRSR are for *one-half* sized images.

Here are some notations used in the following text to indicate the variants of proposed method. **DRSR-Dir** for depth reconstruction and its super-resolution using direct approach, **DRSR-Dir-BF** for depth reconstruction and its super-resolution using direct approach with bilateral filter as an end module in the SR pipeline, similarly for **DRSR-Hier** and **DRSR-Hier-BF** but using hierarchical approach, **Denoising-MFill** for denoising method using median filling approach, and **Inpainting-MFill** and **Inpainting-PFit** for inpainting method using median filling and plane fitting approach respectively.

This section show the results for DRSR on sparse LR depth images with 50% and 10% visible depth points which need to be super-resolved by factor $\times 2$ and $\times 4$. Figure 5.10 shows the output of DRSR on sparse LR depth image with 50% visible data which is super-resolved by factor $\times 2$ and $\times 4$ with direct and hierarchical approach. Similarly, Figure 5.12 shows the output of DRSR on sparse LR depth image with 10% visible data. The 10% scenario is more challenging, because, there are very few visible pixels in the LR image.

The DRSR results are compared with its variants (i.e. DRSR-Dir-MFil, DRSR-Hier-MFil, DRSR-Dir-PFit and DRSR-Hier-PFit) which uses MS or SLIC segment cues and also with a recent work of super-resolution from under sampled data (Mandal et al., 2017), and this method is referred here as Depth map Restoration from Undersample data (SR-DRU).

Figure 5.10 shows the output of DRSR on LR image with 50% visible data which is super-resolved by factor $\times 2$ and $\times 4$, and for better visualization a small region cropped and zoomed is shown in Figure 5.11. In both the cases, the SR results are obtained using MS and SLIC segment cues, using both direct and hierarchical approaches. The first-row of Figure 5.10 shows the results of DRSR method for upsampling factor $\times 2$ on noiseless LR image with 50% visible data. The results of proposed method are comparable to the results shown in SR-DRU column (Mandal et al., 2017) and it is seen that the results of SR-DRU method show some artifacts at the edge discontinuities, whereas the proposed method (either using MS or SLIC segment cues), preserve the edge discontinuities much better. The second-row in Figure 5.10 shows results for DRSR upsampled by factor $\times 4$ on noiseless LR with 50% visible data. Here also the proposed method show plausibly good outputs with sharp edges discontinuities. For better visualization, a small regions of the output image shown in Figure 5.10 is cropped and zoomed and it is shown in Figure 5.11.

Similarly, Figure 5.12 shows the results of DRSR varients and it is compared with SR-DRU method for even more challenging situation where the LR input image is with 10% visible data only. It shows results for upsampling factor $\times 2$ and $\times 4$ for both noiseless and noisy images. The results of the proposed method for this type of LR images are much better than the results of SR-DRU method. For better visual representation, a small region is cropped and zoomed, which is shown in Figure 5.13.

Table 5.3, 5.4 and 5.5 shows quantitative measure of DRSR on LR images with 50%, 10% and 5% visible data respectively. The PSNR and SSIM metrics are computed for the outputs obtained by the proposed DRSR method are compared with the state-of-the-art method of Depth map Restoration from Undersampled data (SR-DRU) by Mandal et al. (2017) on few depth images taken from Middlebury dataset. It can be seen Table 5.3 that for upsampling factor $\times 2$ with 50% visible pixels in LR image, the

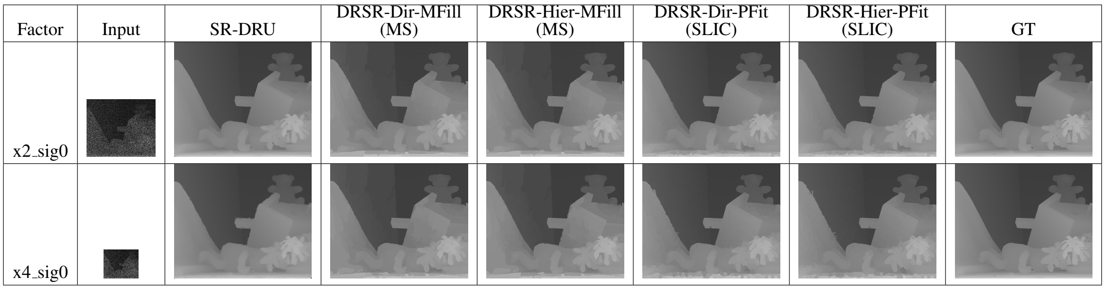


Figure 5.10: DRSR results for upsampling factor ×2 and ×4 from sparse LR image with 50% visible depth pixels

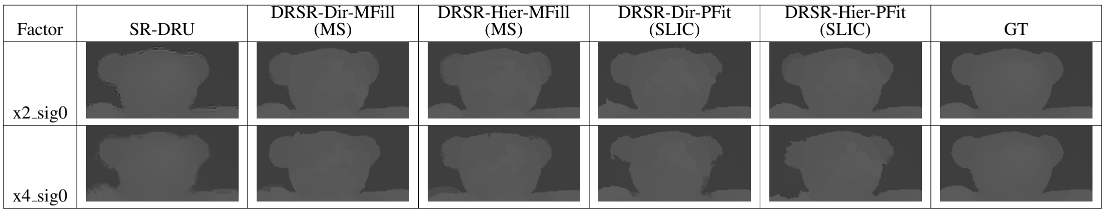


Figure 5.11: Cropped region of images from Figure 5.10

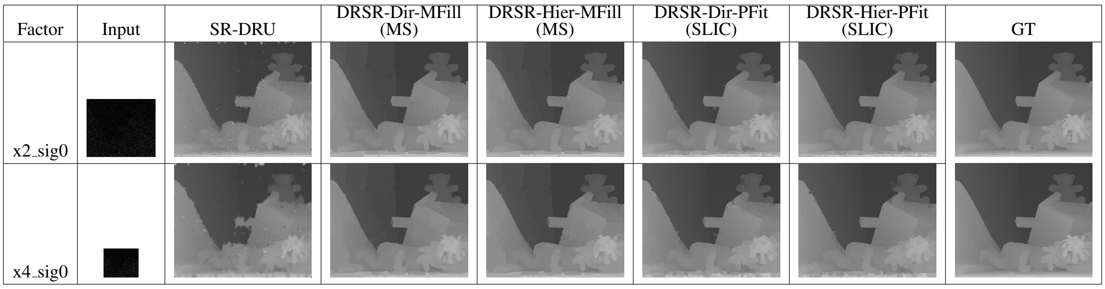


Figure 5.12: DRSR results for upsampling factor ×2 and ×4 from sparse LR image with 10% visible depth pixels

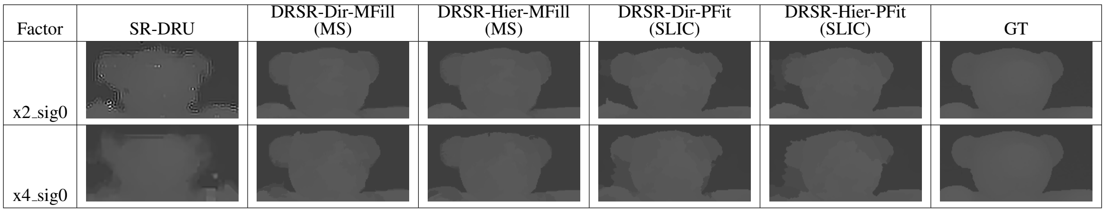


Figure 5.13: Cropped region of images from Figure 5.12

variants of proposed DRSR method perform better as compared to SR-DRU method. However, for upsampling factor ×4, the proposed DRSR method produce outputs close to the SR-DRU method. The 10% data is very less to perform the dense depth reconstruction and its super-resolution.

Table 5.4 shows the results of DRSR for 10% visible pixels in LR image. The variants of proposed method perform well in terms of edge preservation and retaining the overall structure of the objects in the scene, and show good performance as compared to SR-DRU method. The experiments have also been performed with even lower percentage of visible pixel for DRSR problem, i.e. for 5% of visible data, and the results are shown in Table 5.5. The results of SR-DRU degrade heavily because the dictionary learning based methods does not provide much information for image reconstruction.

The results in Table 5.3, 5.4 and 5.5 are for the MS or SLIC based segmentation method only with the MFill approach (i.e. DRSR-Dir, DRSR-Dir-BF, DRSR-Hier and DRSR-Hier-BF). As it was observed in earlier experiments of depth reconstruction that the PFit approach also perform well, hence the SR results using PFit approach for variants of DRSR method using MS segment cues for upsamping factor $\times 2$ and $\times 4$ from 50%, 10% and 5% sparse LR depth pixels are shown in Table 5.6. It is observed that the results produced by the proposed DRSR method is better than SR-DRU method for most of the images. For upsampling factor $\times 4$ with LR image having only 50% visible pixels, the SR-DRU results are comparable. As the number of visible pixels goes lower to 10% and 5%, the proposed variants of DRSR method performs better.

Table 5.3: PSNR/SSIM results of DRSR for upsampling factor $\times 2$ and $\times 4$ from sparse LR image with 50% visible depth point using MS and SLIC segment cues with MFill approach. Notation $\times i\_\mathrm{sig}j$ indicates SR for upsampling factor i on images with noise standard deviation j. **First best results in bold**

SR Factor	Test Images	SR-DRU	MFill (MS)				MFill (SLIC)			
			DRSR-Dir	DRSR-Dir-BF	DRSR-Hier	DRSR-Hier-BF	DRSR-Dir	DRSR-Dir-BF	DRSR-Hier	DRSR-Hier-BF
x2_sig0	Aloe	25.50/0.91	29.87/0.94	30.49/0.95	29.82/0.94	30.44/0.95	31.48/0.95	**32.43/0.96**	31.29/0.95	32.22/0.96
	Art	23.49/0.87	31.04/0.95	**31.58/0.95**	30.80/0.95	31.39/0.95	29.39/0.93	30.02/0.93	28.97/0.92	29.62/0.93
	Baby	30.33/0.96	32.18/0.97	32.92/0.98	32.04/0.97	32.80/0.98	35.47/0.98	**36.69/0.98**	35.10/0.98	36.29/0.98
	Books	32.51/0.97	28.19/0.94	28.56/0.95	28.20/0.94	28.60/0.95	36.46/0.97	**37.50/0.98**	36.40/0.97	37.44/0.98
	Bowling	27.18/0.95	20.79/0.91	20.97/0.91	20.77/0.91	20.95/0.91	31.55/0.97	**32.73/0.97**	31.19/0.97	32.31/0.97
	Cones	30.62/0.99	34.98/0.98	35.69/0.98	34.91/0.98	35.64/0.98	35.60/0.98	**36.62/0.99**	35.45/0.98	36.44/0.98
	Moebius	33.20/0.97	35.76/0.97	36.40/0.98	35.75/0.97	36.38/0.98	37.05/0.97	**38.14/0.98**	36.69/0.97	37.76/0.98
	Plastic	30.53/0.97	30.29/0.99	30.31/0.99	30.14/0.99	30.27/0.99	40.04/0.99	**41.27/0.99**	39.09/0.99	40.24/0.99
	Reindeer	26.83/0.94	31.26/0.96	31.77/0.97	31.18/0.96	31.66/0.97	32.43/0.96	**33.45/0.97**	31.81/0.96	32.79/0.96
	Teddy	33.51/0.99	35.62/0.98	36.21/0.98	35.52/0.98	36.14/0.98	37.36/0.98	**38.57/0.99**	36.90/0.98	38.05/0.99
x4_sig0	Aloe	**31.63/0.95**	29.32/0.93	29.88/0.93	28.88/0.92	29.39/0.93	26.49/0.87	26.91/0.88	26.05/0.87	26.43/0.88
	Art	28.89/0.92	29.40/0.93	**29.90/0.93**	28.76/0.92	29.30/0.93	24.92/0.82	25.34/0.83	24.84/0.83	25.23/0.84
	Baby	**36.19/0.98**	30.04/0.97	30.55/0.97	29.97/0.96	30.51/0.97	32.37/0.95	33.14/0.96	31.99/0.95	32.70/0.96
	Books	**36.70/0.98**	28.16/0.94	28.46/0.95	28.12/0.93	28.46/0.94	32.14/0.94	32.70/0.95	31.91/0.94	32.47/0.95
	Bowling	**32.77/0.98**	20.14/0.90	20.30/0.90	19.74/0.89	19.89/0.90	28.24/0.93	28.99/0.94	28.07/0.93	28.82/0.94
	Cones	**35.15/0.98**	34.01/0.97	34.73/0.98	33.67/0.97	34.35/0.97	32.21/0.96	32.83/0.96	32.36/0.96	32.95/0.96
	Moebius	**36.64/0.98**	34.42/0.97	35.04/0.97	34.22/0.96	34.88/0.97	33.06/0.94	33.69/0.94	32.84/0.94	33.44/0.94
	Plastic	**37.28/0.99**	30.28/0.98	30.41/0.99	30.21/0.98	30.38/0.98	34.44/0.97	35.18/0.97	33.72/0.97	34.41/0.97
	Reindeer	3**2.03/0.97**	30.40/0.96	30.90/0.96	29.91/0.95	30.43/0.96	28.14/0.91	28.72/0.92	27.99/0.91	28.53/0.92
	Teddy	**37.56/0.99**	35.34/0.98	35.90/0.98	34.88/0.97	35.50/0.98	34.05/0.97	34.80/0.97	34.25/0.97	34.99/0.97

Table 5.4: PSNR/SSIM results of DRSR for upsampling factor ×2 and ×4 from sparse LR image with 10% visible depth point using MS and SLIC segment cues with MFill approach. Notation $\times i$_sig$j$ indicates SR for upsampling factor i on images with noise standard deviation j. **First best results in bold**

SR Factor	Test Images	SR-DRU	MFill (MS)				MFill (SLIC)			
			DRSR-Dir	DRSR-Dir-BF	DRSR-Hier	DRSR-Hier-BF	DRSR-Dir	DRSR-Dir-BF	DRSR-Hier	DRSR-Hier-BF
x2_sig0	Aloe	21.21/0.79	29.54/0.94	30.11/0.94	29.43/0.94	30.00/0.94	30.60/0.94	**31.39/0.95**	30.24/0.94	30.99/0.94
	Art	19.07/0.66	30.42/0.94	30.91/0.94	30.22/0.94	**30.75/0.94**	28.40/0.90	28.92/0.91	28.21/0.90	28.75/0.91
	Baby	24.25/0.83	32.10/0.97	32.82/0.98	31.96/0.97	32.68/0.97	34.61/0.97	**35.63/0.97**	34.15/0.97	35.10/0.97
	Books	24.56/0.86	28.25/0.94	28.59/0.95	28.25/0.94	28.62/0.95	35.53/0.97	36.38/0.97	35.63/0.97	**36.50/0.97**
	Bowling	21.66/0.81	20.75/0.90	20.92/0.91	20.73/0.90	20.90/0.91	30.51/0.96	**31.50/0.96**	30.26/0.96	31.17/0.96
	Cones	24.38/0.90	34.57/0.98	35.26/0.98	34.47/0.98	35.18/0.98	35.04/0.98	**35.84/0.98**	34.79/0.98	35.57/0.98
	Moebius	24.58/0.82	35.09/0.97	35.72/0.97	35.09/0.97	35.68/0.97	35.99/0.96	**36.89/0.97**	35.87/0.96	36.77/0.97
	Plastic	23.65/0.77	30.14/0.99	30.17/0.99	30.01/0.98	30.14/0.99	38.68/0.99	**39.59/0.99**	38.01/0.99	38.91/0.99
	Reindeer	21.84/0.80	30.65/0.96	31.14/0.96	30.61/0.96	31.09/0.96	31.47/0.95	**32.28/0.96**	30.97/0.95	31.75/0.95
	Teddy	24.14/0.86	35.29/0.98	35.89/0.98	35.17/0.98	35.79/0.98	36.44/0.98	**37.45/0.98**	36.14/0.98	37.13/0.98
x4_sig0	Aloe	26.40/0.87	28.33/0.91	**28.85/0.92**	28.02/0.91	28.53/0.92	25.80/0.85	26.14/0.86	25.47/0.85	25.78/0.86
	Art	24.46/0.79	26.98/0.89	**27.40/0.89**	26.88/0.89	27.29/0.89	24.31/0.79	24.66/0.80	24.29/0.80	24.64/0.81
	Baby	30.31/0.93	29.89/0.96	30.33/0.97	29.80/0.96	30.27/0.97	31.11/0.94	**31.67/0.95**	30.52/0.94	31.04/0.94
	Books	30.27/0.93	27.94/0.94	28.23/0.94	27.79/0.93	28.13/0.94	31.47/0.93	**31.94/0.94**	31.16/0.93	31.63/0.94
	Bowling	26.26/0.91	20.11/0.90	20.28/0.90	20.15/0.90	20.32/0.90	26.98/0.92	**27.59/0.93**	26.96/0.92	27.59/0.93
	Cones	29.94/0.93	33.35/0.97	33.98/0.97	33.38/0.97	**34.00/0.97**	31.71/0.95	32.23/0.95	31.77/0.95	32.24/0.95
	Moebius	29.74/0.91	33.30/0.96	**33.87/0.96**	33.08/0.96	33.66/0.96	31.28/0.92	31.75/0.93	31.01/0.92	31.46/0.93
	Plastic	30.76/0.93	30.27/0.98	30.41/0.99	30.23/0.98	30.42/0.99	33.49/0.96	**34.10/0.97**	33.20/0.96	33.78/0.97
	Reindeer	27.48/0.90	29.71/0.95	**30.22/0.96**	29.39/0.95	29.87/0.95	27.43/0.89	27.93/0.90	27.36/0.89	27.82/0.90
	Teddy	30.17/0.93	34.57/0.97	**35.06/0.98**	34.17/0.97	34.71/0.97	33.05/0.96	33.66/0.96	33.08/0.96	33.68/0.96

Table 5.5: PSNR/SSIM results of DRSR for upsampling factor ×2 and ×4 from sparse LR image with 5% visible depth point using MS and SLIC segment cues with MFill approach. Notation $\times i$_sig$j$ indicates SR for upsampling factor i on images with noise standard deviation j. **First best results in bold**

SR Factor	Test Images	SR-DRU	MFill (MS)				MFill (SLIC)			
			DRSR-Dir	DRSR-Dir-BF	DRSR-Hier	DRSR-Hier-BF	DRSR-Dir	DRSR-Dir-BF	DRSR-Hier	DRSR-Hier-BF
x2_sig0	Aloe	19.41/0.74	29.00/0.93	29.57/0.93	28.94/0.93	29.50/0.93	29.21/0.92	**29.80/0.93**	29.03/0.92	29.62/0.93
	Art	17.30/0.58	29.46/0.93	**29.92/0.93**	29.24/0.93	29.71/0.93	27.29/0.88	27.78/0.89	27.16/0.88	27.65/0.88
	Baby	22.31/0.82	31.94/0.97	32.65/0.97	31.78/0.97	32.50/0.97	33.39/0.96	**34.23/0.97**	32.94/0.96	33.72/0.96
	Books	21.13/0.81	28.19/0.94	28.54/0.95	28.18/0.94	28.55/0.95	34.95/0.96	**35.71/0.97**	34.85/0.96	35.62/0.97
	Bowling	18.55/0.74	20.75/0.90	20.92/0.91	20.74/0.90	20.90/0.91	29.41/0.95	**30.24/0.96**	29.07/0.95	29.83/0.95
	Cones	21.83/0.86	33.94/0.97	34.61/0.97	33.83/0.97	34.50/0.97	34.22/0.97	34.89/0.98	34.26/0.97	**34.94/0.97**
	Moebius	21.12/0.78	34.92/0.97	35.57/0.97	34.90/0.97	35.49/0.97	35.14/0.96	35.87/0.96	34.93/0.96	**35.66/0.96**
	Plastic	21.66/0.76	30.12/0.99	30.15/0.99	30.00/0.98	30.13/0.99	35.42/0.98	**36.15/0.98**	35.24/0.98	35.94/0.98
	Reindeer	20.08/0.75	30.37/0.95	30.89/0.96	30.28/0.95	30.79/0.96	30.38/0.94	31.08/0.95	30.14/0.94	**30.84/0.94**
	Teddy	21.79/0.85	35.00/0.98	35.59/0.98	34.88/0.98	35.49/0.98	35.90/0.98	36.81/0.98	35.73/0.98	**36.63/0.98**
x4_sig0	Aloe	24.32/0.83	27.52/0.90	**27.96/0.91**	27.16/0.90	27.56/0.90	25.01/0.84	25.29/0.85	24.99/0.84	25.27/0.85
	Art	22.08/0.73	25.09/0.86	**25.40/0.87**	24.95/0.86	25.26/0.86	23.00/0.77	23.26/0.78	22.90/0.77	23.15/0.78
	Baby	28.58/0.91	29.17/0.96	29.52/0.96	28.98/0.96	29.35/0.96	29.19/0.93	**29.55/0.93**	28.64/0.92	28.98/0.93
	Books	26.86/0.91	27.62/0.93	27.91/0.94	27.54/0.93	27.87/0.93	30.79/0.93	**31.19/0.93**	29.96/0.92	30.31/0.93
	Bowling	22.70/0.87	20.08/0.89	20.24/0.90	19.94/0.89	20.09/0.90	24.82/0.89	**25.22/0.90**	24.40/0.89	24.78/0.90
	Cones	25.92/0.90	31.86/0.96	32.35/0.96	31.90/0.96	**32.37/0.96**	30.45/0.94	30.87/0.94	30.58/0.94	30.97/0.94
	Moebius	27.07/0.89	31.82/0.95	**32.24/0.96**	31.72/0.95	32.11/0.96	29.21/0.91	29.54/0.92	29.00/0.91	29.32/0.92
	Plastic	27.67/0.91	30.19/0.98	30.33/0.99	30.17/0.98	30.34/0.99	30.71/0.95	**31.11/0.95**	30.23/0.94	30.60/0.95
	Reindeer	24.77/0.86	28.61/0.94	**29.10/0.95**	28.14/0.94	28.65/0.94	26.31/0.88	26.73/0.89	26.03/0.88	26.41/0.89
	Teddy	26.15/0.90	33.25/0.96	**33.69/0.97**	32.94/0.96	33.43/0.96	31.85/0.95	32.32/0.96	31.64/0.95	32.09/0.96

5.5.3 Results of Depth Image Denoising

This section provide the results for the adaptation of the proposed approach for the task of depth denoising. Same dataset of Middlebury (Scharstein and Szeliski, 2002) has been used for experimentation. For simulating the noisy scenario, additive Gaussian noise with different standard deviation, i.e. 1, 2, 3, 4, 5 and 10, is added to the ground truth image, and their denoised results are presented here.

Table 5.6: PSNR/SSIM results of DRSR for upsampling factor $\times 2$ and $\times 4$ from sparse LR image with 50%, 10% and 5% visible depth point using MS segment cues with PFit approach. Notation $\times i\_\mathrm{sig}j$ indicates SR for upsampling factor i on images with noise standard deviation j. **First best results in bold**

SR Factor	Test Images	PFit (MS) (50% of LR)			PFit (MS) (10% of LR)			PFit (MS) (5% of LR)		
		SR-DRU	DRSR-Dir	DRSR-Dir-BF	SR-DRU	DRSR-Dir	DRSR-Dir-BF	SR-DRU	DRSR-Dir	DRSR-Dir-BF
x2_sig0	Aloe	25.50/0.91	28.92/0.96	**29.07/0.96**	21.21/0.79	27.17/0.93	**27.43/0.94**	19.41/0.74	25.20/0.92	**25.40/0.92**
	Art	23.49/0.87	30.28/0.97	**30.45/0.96**	19.07/0.66	30.18/0.94	**30.49/0.94**	17.30/0.58	29.35/0.93	**29.72/0.93**
	Baby	**30.33/0.96**	29.46/0.97	29.73/0.98	24.25/0.83	27.58/0.96	**27.93/0.97**	22.31/0.82	28.54/0.96	**28.96/0.97**
	Books	**32.51/0.97**	24.18/0.95	24.28/0.95	**24.56/0.86**	23.86/0.94	23.99/0.94	21.13/0.81	22.81/0.92	**23.02/0.92**
	Bowling	**27.18/0.95**	15.23/0.84	15.24/0.84	**21.66/0.81**	15.36/0.82	15.38/0.83	**18.55/0.74**	15.62/0.84	15.63/0.84
	Cones	30.62/0.99	36.95/0.99	**37.73/0.99**	24.38/0.90	35.48/0.98	**36.55/0.98**	21.83/0.86	35.12/0.98	**36.12/0.98**
	Moebius	33.20/0.97	36.37/0.98	**36.94/0.98**	24.58/0.82	33.83/0.96	**34.38/0.97**	21.12/0.78	33.34/0.96	**33.91/0.96**
	Plastic	30.53/0.97	36.77/0.99	**36.77/0.99**	23.65/0.77	36.83/0.99	**36.96/0.99**	21.66/0.76	35.97/0.99	**36.16/0.99**
	Reindeer	26.83/0.94	33.07/0.98	**33.27/0.98**	21.84/0.80	28.96/0.96	**30.07/0.97**	20.08/0.75	31.66/0.96	**32.15/0.96**
	Teddy	33.51/0.99	39.15/0.99	**39.45/0.99**	24.14/0.86	36.35/0.99	**37.01/0.99**	21.79/0.85	36.01/0.98	**36.69/0.98**
x4_sig0	Aloe	**31.63/0.95**	28.33/0.94	28.57/0.94	**26.40/0.87**	25.17/0.90	25.36/0.90	24.32/0.83	24.34/0.88	**24.49/0.89**
	Art	28.89/0.92	29.15/0.93	**29.40/0.93**	24.46/0.79	26.49/0.89	**26.81/0.90**	22.08/0.73	25.91/0.88	**26.22/0.88**
	Baby	**36.19/0.98**	28.02/0.96	28.36/0.97	30.31/0.93	25.73/0.95	**25.93/0.95**	**28.58/0.91**	26.45/0.95	26.69/0.95
	Books	**36.70/0.98**	23.98/0.94	24.15/0.94	**30.27/0.93**	22.64/0.91	22.86/0.92	**26.86/0.91**	22.51/0.91	22.72/0.92
	Bowling	**32.77/0.98**	15.16/0.83	15.17/0.84	**26.26/0.91**	15.24/0.83	15.25/0.83	**22.70/0.87**	15.26/0.83	15.27/0.83
	Cones	**35.15/0.98**	33.82/0.98	34.61/0.98	29.94/0.93	32.09/0.97	**32.84/0.97**	25.92/0.90	31.14/0.96	**31.75/0.96**
	Moebius	**36.64/0.98**	33.76/0.97	34.70/0.97	29.74/0.91	31.38/0.95	**32.04/0.95**	27.07/0.89	29.32/0.93	**29.78/0.94**
	Plastic	**37.28/0.99**	36.23/0.99	36.47/0.99	30.76/0.93	36.75/0.99	**37.24/0.99**	27.67/0.91	37.09/0.99	**37.67/0.99**
	Reindeer	**32.03/0.97**	25.56/0.96	26.69/0.97	**27.48/0.90**	24.92/0.95	25.98/0.96	24.77/0.86	25.08/0.94	**26.16/0.95**
	Teddy	37.56/0.99	37.26/0.99	**37.92/0.99**	30.17/0.93	35.34/0.98	**35.92/0.98**	26.15/0.90	33.07/0.97	**33.52/0.97**

Figure 5.14 shows the denoised images. From the results presented in Figure 5.14, it is indeed clearly observed that the proposed guidance based denoising method perform reasonably well. The second and the third-column of Figure 5.14 show the denoising results using MS and SLIC segment cues. As SLIC segmentation generates very local super-pixels, the median filling approach for these super-pixels generates somewhat locally jagged surfaces as compared with MS based denoising.However, in overall, the edge discontinuities are maintained to a larger extent, and the noise has also been reduced.

The quantitative results of denoising compared to other denoising techniques are shown in Table 5.7. The main intension of the proposed denoising method is to show the applicability of the guidance colour image based method for denoising problem. So, the denoising comparison is kept limited to only the popular denoising method i.e. bilateral filter (BF) (Tomasi and Manduchi, 1998), and it is observed that the proposed denoising method is able to maintain a good PSNR value in most of the cases while denoising the image.

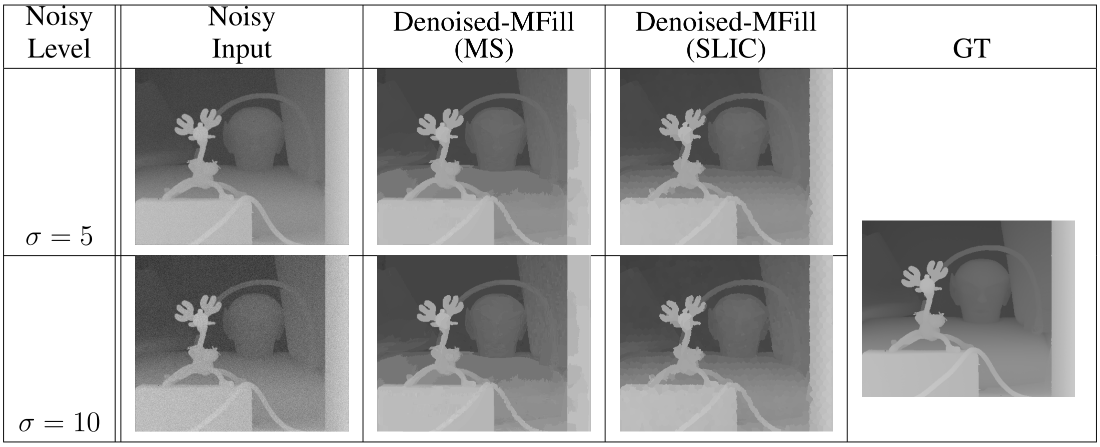


Figure 5.14: Results of depth denoising from noisy images with different noise standard deviation (i.e. $\sigma = 5, 10$)

Table 5.7: PSNR/SSIM results of depth denoising

Noise Level	Test Images	Noisy	BF	Denoised-MFill (MS)	Denoised-MFill (SLIC)
$\sigma = 5$	Aloe	33.69/0.77	35.68/0.96	30.97/0.93	29.98/0.92
	Art	33.09/0.77	33.41/0.94	31.46/0.93	28.25/0.89
	Baby	33.88/0.73	39.38/0.97	33.66/0.97	34.34/0.96
	Books	33.68/0.74	39.73/0.97	31.51/0.96	34.86/0.95
	Cones	33.67/0.76	37.76/0.96	35.48/0.96	33.72/0.95
	Moebius	33.68/0.74	39.99/0.97	37.28/0.96	34.75/0.95
	Plastic	33.72/0.73	39.99/0.97	31.32/0.98	39.61/0.98
	Reindeer	33.41/0.74	36.12/0.96	33.02/0.95	31.78/0.94
	Sawtooth	33.77/0.72	38.72/0.97	39.14/0.98	35.18/0.97
	Teddy	33.72/0.75	39.40/0.97	36.72/0.96	34.42/0.95
	Venus	33.91/0.72	41.57/0.97	36.79/0.97	39.66/0.98
$\sigma = 10$	Aloe	28.08/0.49	34.43/0.91	30.85/0.91	30.08/0.91
	Art	27.83/0.49	32.61/0.89	31.38/0.92	28.36/0.88
	Baby	28.06/0.42	36.82/0.91	33.54/0.96	34.31/0.96
	Books	28.00/0.43	37.02/0.91	31.66/0.95	34.82/0.95
	Cones	28.10/0.48	35.90/0.91	34.98/0.94	33.71/0.94
	Moebius	28.01/0.44	37.16/0.91	36.85/0.95	34.72/0.94
	Plastic	28.01/0.41	37.16/0.91	31.35/0.98	39.19/0.98
	Reindeer	27.93/0.44	34.74/0.91	32.87/0.95	31.80/0.94
	Sawtooth	28.02/0.41	36.46/0.91	38.06/0.96	35.15/0.96
	Teddy	28.07/0.46	36.86/0.92	36.08/0.95	34.32/0.95
	Venus	28.06/0.40	37.87/0.91	36.27/0.95	39.14/0.97

5.5.4 Results of Depth Image Inpainting

This section demonstrate the applicability of the proposed guidance colour image based depth reconstruction method for the problem of depth inpainting. The experiments have been performed on synthetic images (from Middlebury dataset) and real depth images (from Kinect device). The synthetic images are inscribed with various types of missing region, e.g. random missing depth, and kinect-like degraded structured missing depth, and there are also Kinect based captured RGB-D images. The kinect-like degraded images and kinect captured RGB-D images are taken from Yang et al. (2014).

Figure 5.15 shows the results of inpainting by the proposed method, where the first-row shows the input images which need to be painted which has various kinds of missing regions for experimentation purpose. The following rows shows results obtained by variants of proposed method, i.e. Inpaint-MS-MFill, Inpaint-MS-PFit, Inpaint-SLIC-MFill and Inpaint-SLIC-PFit. It is seen that the MS segment cue based method (either MFill or PFit approach) does well in preserving edge discontinuities and the depth precision.

Using SLIC segment cues also, the proposed method does better job of inpainting the missing regions in the synthetic images, however, there are some visible holes at the boundary of the image, but the overall depth precision and object depth discontinuities are mostly preserved. SLIC segmentation generates super-pixels which are finer than the segments generated by MS segmentation, so the MFill approach works well for smoother regions, but at the same time, PFit approach stumble at full retaining the edge discontinuities as seen in the last-row (Inpaint-SLIC-PFit) on kinect captured image.

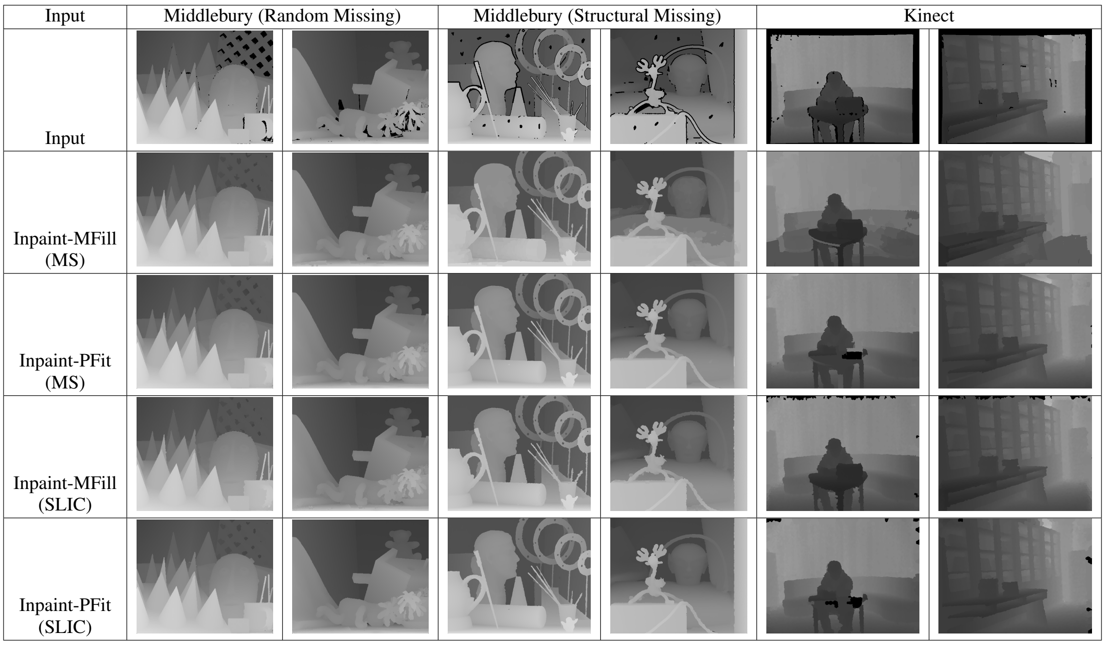


Figure 5.15: Results of depth inpaiting from random missing regions, structural missing regions and real time Kinect images

Another set of experiments were performed for inpainting of the randomly scribbled images. The input images were randomly scribbled to synthesize random missing region in the image. Using the proposed guidance based inpainting method, it could successfully inpaint the scribbled region by preserving the depth edge discontinuities and depth precision at most of the regions. In the last-row of Figure 5.16 and Table 5.8, the scribbles are quite thick but as it is observed that the proposed inpainting methods maintains the overall structure of the scene. Figure 5.16 shows three example of inpainting the random scribble on the depth images. The inpainting performance of the proposed method is shown in Table 5.8 in terms of PSNR/SSIM performance metrics.

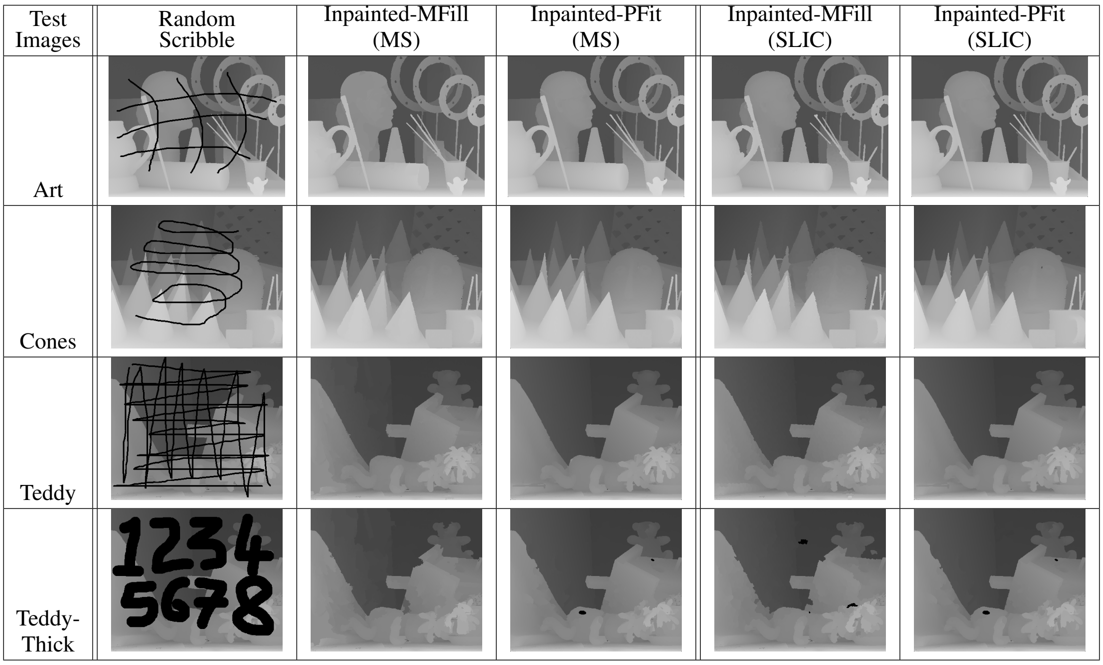


Figure 5.16: Results of depth inpaiting from random scribbled image

Table 5.8: PSNR/SSIM results of inpainting method on random scribbled images

Test Images	Random Scribble	Inpaiting-MFill (MS)	Inpaiting-PFit (MS)	Inpaiting-MFill (SLIC)	Inpaiting-PFit (SLIC)
Art	18.90/0.87	36.52/0.98	39.95/0.99	35.36/0.98	39.95/0.99
Cones	18.49/0.84	41.25/0.99	42.09/0.99	40.42/0.99	40.67/0.99
Teddy	14.40/0.54	38.98/0.98	41.43/0.99	38.88/0.97	41.71/0.99
Teddy Thick	11.47/0.51	32.88/0.96	32.65/0.97	30.16/0.95	32.64/0.97

5.5.5 Failure Scenario

The MS/SLIC segmentation algorithms are sensitive to the low contrast images. When the foreground and background colour contrast is very low, then these segmentation algorithm segment such low contrast region as one single region. In such cases, the segment cue generated will be incorrect, and thus the depth reconstruction or super-resolution gives wrong results.

Figure 5.17 shows the failure case where the the colour contrast of the bowling target and the background is very low (*red* box), and they are segmented as one region which results in the wrong cue for depth image super-resolution. However, in the region of high contrast between foreground and background (*blue* box), the segmentation is accurate and thus the depth reconstruction in reasonably good.

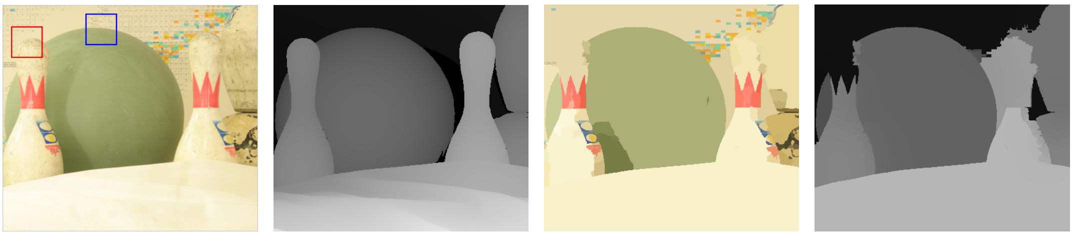

(a) HR colour image (b) HR depth image (c) MS segmented image (d) SR by factor ×2

Figure 5.17: Failure depth super-resolution (*Bowling1* image). Poor performance at segment regions with low-contrast (*red* box), and good performance at segment regions with high-contrast (*blue* box) region

5.6 SUMMARY

Two simplistic and local approaches have been proposed for depth reconstruction from the sparsely sampled random depth data. These methods employ the segmentation cue from colour image of the same scene. The variants of proposed methods are based on locally planar or constant depth assumption and use plane fitting or median computation on local segments, followed by local cost computations. These methods also involve either an iterative process or a 2-step process which reconstructs partial depth maps, and then completes the same. Encouraging qualitative and quantitative results have been demonstrated, and also shown positive comparisons with a state-of-the-art depth reconstruction methods like ADMM and DR-DRU, which are considered as more sophisticated methods.

It is also shown that the proposed DRSR method can be used to super-resolve a sparsely observed LR image by cascading the DR and SR method. For DRSR problem, the input has only few randomly sampled pixels on the LR depth image. This sparse point cloud is fed to DRSR framework, which is a cascade of DR and SR module, which does dense depth reconstruction first, followed by its super-resolution to a desired resolution. For higher upsampling factors, hierarchical approach has also been presented, and comparable results are seen.

The applicability of the proposed guidance colour image based method for depth image denoising and depth image inpainting has been demonstrated, and the results are promising. For denoising problem, various levels of noise have been considered. For Inpainting problem, various types of missing regions on synthetic and real Kinect depth images have been considered. It is observed that SLIC segment cues works better at depth regions which has piecewise linear varying depths.

In guidance image based depth image super-resolution the source of information is from the guided HR colour image and the input LR image only. With recent technology where there is no dearth of computational resources, one can learn the high-frequency information from a set of HR images and use the learned information to obtain the high-frequency detail for an unseen LR input. The next contributory chapter is motivated by abundance of computational resources and the set of LR and its HR image pairs. These LR-HR image pairs are used to learn the mapping between them by training a Gaussian mixture model (GMM) and use the learned model to infer the high-frequency details for a LR test image.

CHAPTER 6

GAUSSIAN MIXTURE MODEL BASED SINGLE DEPTH IMAGE SUPER-RESOLUTION

6.1 INTRODUCTION

[1] As depth image have prominent edges, unlike optical images where texture is also important, so all depth image super-resolution methods try to enhance the edges. In previous chapters, some of the proposed methods for depth image super-resolution problem were seen. These methods were mainly concentrating on enhancing the edges either by using the wavelet transforms to extract the high-frequency information from the input image, or by using an HR guidance colour image to obtain some prior cues to help in refining the image to look plausible or to estimating the unknown pixels on the HR image grid.

There have been few work on training a model to learn the HR-LR relationship from the available training example images. These training based methods work better if there is the training dataset is huge. If the number of training examples were less, it results in overfitting scenario where such a method fails to generalize, however, with large number of training examples the model can learn large types of variations from them. This chapter try to address the problem of single depth image super-resolution using Gaussian Mixture Model (GMM) technique. GMM model has proven to be good for unsupervised clustering based on the probability distribution and it has been widely used in image restoration, clustering and regression problems, among others. For training GMM model, overlapping HR and LR patches generated from synthetic depth images and their downsampled versions respectively, are vectorized and concatenated to

---

[1] Chandra Shaker Balure, Arnav Bhavsar, and M. Ramesh Kini. ”GMM Based Depth Image Super-Resolution.” *National Conference on Computer Vision, Pattern Recognition, Image Processing and Graphics (NCVPRIPG 2017).*

form a training matrix. The inherent relationship between the HR and LR patches are captured by the covariance matrix which helps in estimating the HR patch for the input LR patch. Expectation-Maximization (EM) iterative algorithm was adopted for parameter estimation which guarantees the convergence.

Motivated by the work of Sandeep and Jacob (2016), who have proposed single image SR method for optical images using GMM which learns HR-LR relationship. Inspired by their work, a single depth image super-resolution method using GMM model has been proposed. Depth images have different characteristics as compared to optical images. Mainly, depth images have prominent depth discontinuities, and they lack the texture as most of the region in depth image will be smoother with almost similar values in that region. The standard GMM training and testing procedure remains the same, however the proposed SRGMM method differs by several aspects,

1. GMM model is used for depth image SR as opposed to the optical image SR.
2. Synthetic depth images are used for training purpose as provided by Mac Aodha et al. (2012) which has sharper edges to suit for SR problem.
3. Experiments have been performed with different number of Gaussian mixtures and different patch sizes.
4. A stage-wise GMM training for enabling hierarchical SR is proposed, especially for higher upsampling factors.
5. SR performance has been demonstrated on several unseen depth images from standard Middlebury dataset (Scharstein and Szeliski, 2002), and real ToF depth camera captured depth images by Ferstl et al. (2013) both qualitatively and quantitatively.
6. We show substantial result comparisons with classical bilinear and bicubic interpolation methods and also with other state-of-the-art single depth image SR methods, e.g. guided image filtering SR method by He et al. (2010), anisotropic total generalized variation SR method by Ferstl et al. (2013), and residual interpolation SR method by Konno et al. (2015).

6.2 BACKGROUND

The use of GMM has been well proven to address the problems of speaker recongnition Reynolds et al. (2000), image restoration Portilla et al. (2003), image segmentation/clustering Gupta and Sortrakul (1998); Zivkovic (2004), image super-resolution Sandeep and Jacob (2016) and more. In speaker recognition problem addressed by Reynolds et al. (2000), they have used GMM, universal background model (UBM) and a form of Bayesian adaptation. On the other hand, for image restoration problem, authors in Portilla et al. (2003) have used GMM for image denoising. As GMM is good at clustering the similar pattern under the umbrella of a Gaussian, so it has also been used image segmentation in Gupta and Sortrakul (1998); Zivkovic (2004).

This section presents a brief overview of Gaussian mixture model (GMM) and the expectation-maximization (EM) algorithm to estimate the unknown parameters of the Gaussian distribution.

6.2.1 Gaussian Mixture Model Description

Gaussian Mixture Model (GMM) is a probabilistic model that assumes all the data points are generated from a mixture of a finite number of Gaussian distributions with unknown parameters. One can think of mixture models as generalizing k-means clustering to incorporate information about the covariance structure of the data as well as the centers of the potential Gaussians.

Suppose, the univariate data $\boldsymbol{x} = \{x_1, \cdots, x_n\}$ is the collection of n samples which are independent and identically distributed (i.i.d). Lets us take a simple example of Gaussian parametric distribution to infer the parameters from the data (univariate), and use the model to classify or cluster or generate more data from it. For a underlying example of Gaussian distribution, there are two parameters to be inferred from the given data, one is the mean (μ) and other one is the variance (σ^2), where μ measures the central tendency, and σ^2 measures the variability. Let $\Theta = \{\mu, \sigma^2\}$ be the set of model parameters. Thus, the probability density function (pdf) of such a distribution is given by Eq. 6.1,

$$p(x|\Theta) = \frac{1}{\sqrt{2\pi}\,\sigma} \exp\left(\frac{-(x-\mu)^2}{2\sigma^2}\right) \tag{6.1}$$

By keeping the observations fixed, allow the model parameter to vary and estimate its likelihood, such that, the estimated parameter for the Gaussian is likely to generate those fixed observations. Thus, the maximum-likelihood (ML) estimation is written as a likelihood function as shown in Eq. 6.2,

$$l(\Theta|\boldsymbol{x}) \equiv p(\boldsymbol{x}|\Theta) = \prod_{i=1}^{n} p(x_i|\Theta) \tag{6.2}$$

The log-likelihood of Eq. 6.2 helps in easily optimizing the parameters, because natural logrithm *ln* is a monotonically increasing function, hence, the log-likelihood is defined in Eq. 6.3 as,

$$L(\Theta|\boldsymbol{x}) \equiv \ln\ l(\Theta|\boldsymbol{x}) = \sum_{i=1}^{n} \ln\ p(x_i|\Theta) \tag{6.3}$$

The empirical mean and variance of univariate data, with Gaussian distribution as shown in Eq. 6.1, which are the results of maximizing the log-likelihood $L(\Theta|\boldsymbol{x})$ by solving the partial derivatives with respect to μ and σ, and equating it to zero, is given by $\hat{\mu}$ and $\widehat{\sigma^2}$ as shown in Eq. 6.4 and Eq. 6.5 respectively.

$$\hat{\mu} = \frac{1}{N}\sum_{i=1}^{N} x_i = \frac{1}{N}\boldsymbol{x}^T\mathbf{1} \tag{6.4}$$

$$\hat{\sigma}^2 = \frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2 = \frac{1}{N}(\boldsymbol{x} - \mu\mathbf{1})^T(\boldsymbol{x} - \mu\mathbf{1}) \tag{6.5}$$

Such a model can be further extended for univariate data with multiple Gaussian (K Gaussians) in cases where the single Gaussian cannot represent the data distribution well. Thus, the probability of observation x becomes as shown in Eq. 6.6,

$$\begin{aligned} p(x) &= \sum_{j=1}^{K} P(z = j)\, p(x|z = j) \\ &= \sum_{j=1}^{K} \omega_j \, \frac{1}{\sqrt{2\pi}\,\sigma_j}\ \exp\left(\frac{-(x - \mu_j)^2}{2\sigma_j^2}\right) \end{aligned} \tag{6.6}$$

where, ω_j is the weight of the j^{th} Gaussian, and all the weights must summed to 1, i.e. $\sum_{i=1}^{K} \omega_j = 1$.

For the case of multivariate, which is where is the point of interest, the probability of an element $\overline{x} \in \mathbb{R}^d$ of d dimension is given by Eq. 6.7,

$$p(\overline{x}) = \sum_{j=1}^{K} \omega_j \frac{1}{\sqrt{(2\pi)^d |\boldsymbol{S}_j|}} exp\{-\frac{1}{2}(\overline{x} - \overline{\mu}_j)^T \boldsymbol{S}_j^{-1} (\overline{x} - \overline{\mu}_j)\} \tag{6.7}$$

where, ω_j is the weight, and $\overline{\mu}_j \in \mathbb{R}^d$ is the mean vector of the j^{th} Gaussian, and $\boldsymbol{S}_j$ is the covariance matrix of size $d \times d$ which represents the shape and the orientation of the Gaussian.

To find the maximum-likelihood, expectation-maximization (EM) algorithm is used, which is an iterative method for parameter estimation. A brief explanation of EM algorithm is given in the following section.

6.2.2 Expectation-Maximization Algorithm Description

The main difficulty in learning GMM models from unlabeled data is that, one usually doesn't know which points came from which latent component. For many models, a maximum-likelihood (ML) estimation can be found, but for many other models, there would not be any known closed-form solution to the maximization problem. In such cases, ML estimation has to be found numerically using some optimization methods, which are intractable. The alternate solution is to use EM algorithm.

Expectation-maximization (EM) is a well-founded statistical algorithm to get around this problem by an iterative process. The EM algorithm iterates between expectation-step (E-step) and maximization-step (M-step). In E-step, the log-likelihood is *evaluated* based on the current or initially set parameters, and in M-step, it *updates* the parameters by maximizing the expected log-likelihood found in E-step. E-step assumes random components (randomly centered on data points, or learned from k-means, or even just normally distributed around the origin), and computes for each point a probability of being generated by each component of the model. Then, in second step (M-step), one tweaks the parameters to maximize the likelihood of the data given those assignments. Repeating this process is guaranteed to always converge to a local optimum. The summary of EM algorithm is as below:

1. Initialize the weights (ω_j), means (μ_j), and variances (σ_j^2) for each Gaussian in the model.
2. Calculate the posterior probability that these Gaussians have generated the data collected in the dataset.
3. Update the weights, means and variances for each Gaussian.
4. Repeat step 2 and 3 until convergence.

6.3 PROPOSED LEARNING BASED GMM MODEL FOR DEPTH IMAGE SR

Let us denote the following notations for better understanding. We denote the HR and LR training image set as $\boldsymbol{X}$ and $\boldsymbol{Y}$ respectively, with T number of training examples in each set, i.e. $[x_1, \cdots, x_T]$ and $[y_1, \cdots, y_T]$ respectively. We extract N number of patch-pair (HR-LR patch-pair) from the given HR and LR training set $\boldsymbol{X}$ and $\boldsymbol{Y}$, and call the set of HR and LR patches as $\boldsymbol{PX}$ and $\boldsymbol{PY}$ respectively, where each set containing N number of patches $[px_1, \cdots, px_N]$ and $[py_1, \cdots, py_N]$. The patches px_i extracted from an HR training image x_i is of size $q\tau \times q\tau$, and the patches py_i from an LR training image y_i is of size $\tau \times \tau$, where q is the upsampling factor. These HR and LR set of patches $\boldsymbol{PX}$ and $\boldsymbol{PY}$ are then converted into vector form to form a set of HR and LR vectors which is represented as $\overline{\boldsymbol{VX}}$ and $\overline{\boldsymbol{VY}}$, where each set is represented by $[\overline{vx}_1, \cdots, \overline{vx}_N]$ and $[\overline{vy}_1, \cdots, \overline{vy}_N]$ respectively. These HR and LR vectors are concatenated to form a single concatenated vector, and the complete set of such N vectors $[\overline{v}_1, \cdots, \overline{v}_N]$ is represents as $\overline{\boldsymbol{V}}$.

With this nomenclature, for a given image set $\boldsymbol{X}$ and $\boldsymbol{Y}$ we train the GMM with the concatenated HR-LR patch vector set $\overline{\boldsymbol{V}}$. For an unseen LR input image y, the problem of SR involves upsampling y by a factor of q to produce an HR estimate $\hat{x}$, which needs to be closer to the ground truth (GT) image x.

6.3.1 SR GMM Training

Let us assume that we have a set HR and LR training image set $\boldsymbol{X} = \{x_i\}_{i=1}^{T}$ and $\boldsymbol{Y} = \{y_i\}_{i=1}^{T}$ respectively, with T numbers of training examples in each set. The LR image set $\boldsymbol{X}$ is the downsampled versions of the HR image set $\boldsymbol{Y}$. We first extract total N overlapping patches from each HR and LR training set. The patches extracted from HR image set $\boldsymbol{X}$ are of size $q\tau \times q\tau$, and from LR image set $\boldsymbol{Y}$ are of size $\tau \times \tau$, and they are represented by $\boldsymbol{PX} = \{px_i\}_{i=1}^{N}$ and $\boldsymbol{PY} = \{py_i\}_{i=1}^{N}$ respectively. These patches are then converted into vector form by stretching the patches into a column vector, and it is represented by $\overline{\boldsymbol{VX}} = \{\overline{vx}_i\}_{i=1}^{N}$, where $\overline{vx}_i \in \mathbb{R}^{\tau^2 q^2}$, and $\overline{\boldsymbol{VY}} = \{\overline{vy}_i\}_{i=1}^{N}$, where $\overline{vy} \in \mathbb{R}^{\tau^2}$, for HR and LR patch set respectively. These HR and LR vector set are concatenated, as shown in Eq. 6.8, to form a single matrix of width equal to the number of vectors (i.e. N), and height equal to the addition of length of HR and LR vector (i.e. $\tau^2 q^2 + \tau^2$), where each column vector is represented by $\overline{v_i} \in \mathbb{R}^{\tau^2(1+q^2)}$ (or $\mathbb{R}^d$, where $d = \tau^2(1+q^2)$), and the collection of all such N vectors is represented by $\overline{\boldsymbol{V}} = \{\overline{v}_i\}_{i=1}^{N}$. The vector set $\overline{\boldsymbol{V}}$ is the observation matrix for the GMM training.

$$\overline{v}_i = \begin{bmatrix} \overline{vx}_i \\ \overline{vy}_i \end{bmatrix} \tag{6.8}$$

For training GMM, parameter K, the number of Gaussian components, needs to be specified. GMM prior is a mixture of K Gaussian components with parameters $\{\overline{\mu}_j, \boldsymbol{S}_j\}_{j=1}^{K}$. A randomly chosen vector $\overline{z}$, which represents the concatenated HR and LR patch vector will have a probability density function (pdf) as shown in Eq. 6.9,

$$p(\overline{z}) = \sum_{j=1}^{K} \omega_j \; \Phi\left(\overline{z}; \overline{\mu}_j, \boldsymbol{S}_j\right) \tag{6.9}$$

where, $\overline{\mu}_j$ and $\boldsymbol{S}_j$ denote the mean vector and covariance matrix of the j^{th} Gaussian mixture respectively, ω_j is its weight, and the function $\Phi(\cdot)$ denotes multivariate Gaussian pdf which is given as below,

$$\Phi(\overline{z};\overline{\mu}_j,\boldsymbol{S}_j)=\frac{1}{\sqrt{(2\pi)^d|\boldsymbol{S}_j|}}\ \exp\left(-\frac{1}{2}(\overline{z}-\overline{\mu}_j)^T\boldsymbol{S}_j^{-1}(\overline{z}-\overline{\mu}_j)\right) \tag{6.10}$$

The whole vector space $\{\overline{v}_i\}$ grouped by GMM is completely characterized by the parameter set $\Theta=\{\omega_j,\overline{\mu}_j,\boldsymbol{S}_j\}$, where $\overline{z},\overline{\mu}_j\in\mathbb{R}^d$ and $\boldsymbol{S}_j\in\mathbb{R}^{d\times d}$

The likelihood of a vector $\overline{v}_i$ belonging to the j^{th} Gaussian is denoted by a random variable (RV) r, which can take on value $j=1,\cdots,K$, which corresponds to the Gaussian which generated it. This likelihood is given by Eq. 6.11,

$$\begin{aligned}q_i^{(j)}&\equiv P(r_i=j|\overline{v}_i)\\&=P(r_i=j)\ p(\overline{v}_i|r_i=j)\\&=\omega_j\ p(\overline{v}_i|r_i=j)\\&=\omega_j\ \Phi(\overline{v_i};\overline{\mu}_j,\boldsymbol{S}_j)\\&=\omega_j\ \frac{1}{\sqrt{(2\pi)^d|\boldsymbol{S}_j|}}\ \exp\left(-\frac{1}{2}(\overline{v}_i-\overline{\mu}_j)^T\boldsymbol{S}_j^{-1}(\overline{v}_i-\overline{\mu}_j)\right)\end{aligned} \tag{6.11}$$

By maximizing the likelihood $q_i^{(j)}$ of i^{th} vector $\overline{v}_i$, in j^{th} Gaussian is as shown in Eq. 6.12,

$$\hat{j}_i=arg\ \underset{j}{max}\ q_i^{(j)} \tag{6.12}$$

The j^{th} Gaussian component over some vectors $\overline{v}_i$ can be treated as grouping of similar vectors, such that, the HR part of these concatenated vector with similar behavior is grouped in a cluster in their HR patch space, and similarly the LR part of the vector gets grouped in their LR patch space. The parameter $\bar{\mu}_j$ and $\boldsymbol{S}_j$ of the j^{th} Gaussian mixture can be represented by Eq. 6.13 and Eq. 6.14 respectively,

$$\overline{\mu}_j=\begin{bmatrix}\overline{\mu}_{H_j}\\ \overline{\mu}_{L_j}\end{bmatrix} \tag{6.13}$$

$$\boldsymbol{S}_j=\begin{bmatrix}\boldsymbol{S}_{H_j} & \boldsymbol{S}_{HL_j}\\ \boldsymbol{S}_{LH_j} & \boldsymbol{S}_{L_j}\end{bmatrix} \tag{6.14}$$

where, $\overline{\mu}_{(\cdot)}$ and $\boldsymbol{S}_{(\cdot)}$ represent the mean vectors and the covariance matrices. The subscript H_j and L_j corresponds to the HR and LR part, such that S_{H_j} and S_{L_j} represents the covariance matrices between the HR vectors in HR vector space and between LR vectors in LR vector space respectively, of j^{th} Gaussian mixture; similarly, the subscript S_{HL_j} corresponds to the covariance matrix with its elements as covariance between HR and LR vectors (and S_{LH_j} is transpose of S_{HL_j}). Each Gaussian mixture represents a space which contains the HR-LR pair whose mean and covariance is close to the Gaussian mixture it belongs to.

For image SR problem, the cross covariance matrix $\boldsymbol{S}_{HL_j}$ is utilized to estimate the HR patch corresponding to the LR patch of the input test image. Given a set of vectors $\overline{v}_i = [\overline{v}_1, \cdots, \overline{v}_N]$, the parameter $\Theta = \{\omega_j, \overline{\mu}_j, \boldsymbol{S}_j\}_{j=1}^{K}$ is learnt by maximizing the likelihood of the data. With a given initial parameter $\tilde{\Theta} = \{\tilde{\omega}_j, \tilde{\overline{\mu}}_j, \tilde{\boldsymbol{S}}_j\}$, the objective function for ML estimation is given by Eq. 6.15,

$$\begin{aligned}\Theta &= \arg\max_{\tilde{\Theta}} p(\overline{v}_1, \cdots, \overline{v}_N|\tilde{\Theta}) \\ &= \arg\max_{\tilde{\Theta}} \; -\sum_{i=1}^{N} \log \sum_{j=1}^{K} \tilde{\omega}_j \mathcal{N}(\overline{v}_i; \tilde{\overline{\mu}}_j, \tilde{\boldsymbol{S}}_j)\end{aligned} \tag{6.15}$$

The ML estimation is difficult and it does not give exact solution, thus EM algorithm is used to compute the GMM parameters which guarantee the convergence. As discussed earlier, the EM algorithm iterates alternatively between E-step (expectation) and M-step (maximization) to update the parameters until convergence.

Algorithm 7 shows the complete process of training the GMM model, and its parameter estimation.

6.3.2 Multivariate GMM Testing

As a part of testing the GMM model for SR performance, the input test image is decomposed into overlapping patches of size $\tau \times \tau$. All patches are than converted into vector form which are represented by $\{\overline{y}_i\}_{i=1}^{N}$, where $\overline{y}_i \in \mathbb{R}^{\tau^2}$. Given a patch from the test

Algorithm 7 Multivariate GMM training

```
1: Input: Set of training vectors $\{\overline{v}_i\}_{i=1}^{N}$, number of Gaussian components ($K$), and Threshold ($\delta$).
2: Initialization: The parameters for multivariate GMM $\{\omega_j, \overline{\mu}_j, \boldsymbol{S}_j\}_{j=1}^{K}$ are initialized by randomly partitioning the $\overline{v}_i$'s into $K$ clusters $\{\mathbb{C}_j\}_{j=1}^{K}$, and compute $\omega_j, \overline{\mu}_j$ and $\boldsymbol{S}_j$ as,
       $\omega_j = \frac{|\mathbb{C}_j|}{N};$
       $\overline{\mu}_j = \frac{1}{\mathbb{C}_j} \sum_{m \in \mathbb{C}_j} \overline{v}_m;$
       $\boldsymbol{S}_j = \frac{1}{\mathbb{C}_j} \sum_{m \in \mathbb{C}_j} (\overline{v}_m - \overline{\mu}_j)(\overline{v}_m - \overline{\mu}_j)^T$
3: E-Step:
       $q_i^{(j)} = \frac{\omega_j \; \Phi(\overline{v_i}; \overline{\mu}_j, \boldsymbol{S}_j)}{\sum_{j=1}^{K} \omega_j \; \Phi(\overline{v_i}; \overline{\mu}_j, \boldsymbol{S}_j)}$
   and
       $n_j = \sum_{i=1}^{N} q_i^{(j)}$
   where, $i = 1, \cdots, N$ is the number of samples collected, and $j = 1, \cdots, K$ is the number of Gaussian components.
4: M-Step:
       $\omega_j = \frac{n_j}{N};$
       $\overline{\mu}_j = \frac{1}{n_j} \sum_{i=1}^{N} q_i^{(j)} \overline{v}_i;$
       $\boldsymbol{S}_j = \frac{1}{n_j} \sum_{i=1}^{N} q_i^{(j)} (\overline{v}_i - \overline{\mu}_j)(\overline{v}_i - \overline{\mu}_j)^T$
5: Convergence criterion: Compute the likelihood $\hat{\mathbb{L}}$, such that,
       $\hat{\mathbb{L}} = \frac{1}{N} \sum_{i=1}^{N} \log \left( \sum_{j=1}^{K} \omega_j \; \Phi(\overline{v_i}; \overline{\mu}_j, \boldsymbol{S}_j) \right)$
   If $|\mathbb{L} - \hat{\mathbb{L}}| < \delta$;    goto step-6
       Otherwise;    set $\mathbb{L} = \hat{\mathbb{L}}$, and goto step-3
6: Output: GMM parameters $\{\omega_j, \overline{\mu}_j, \boldsymbol{S}_j\}_{j=1}^{K}$ for $K$ Gaussian components.
```

image, the likelihood of that patch being generated from a Gaussian component (say, j^{th} Gaussian component) from the set of Gaussian components is estimated is given by Eq. 6.16 as,

$$\gamma_{L_j}^{i} = \omega_j \; p(\overline{y}_i | \overline{\mu}_{L_j}, \boldsymbol{S}_{L_j}) \tag{6.16}$$

and choose the one which maximizes the likelihood, which is given by Eq. 6.17,

$$\hat{j}_i = \underset{j}{arg\; max}\; \gamma_{L_j}^{i} \tag{6.17}$$

On estimating the Gaussian component which could possibly be responsible for generating the test vector, we now estimate the corresponding HR patch $\hat{x}_i$ from the j^{th} Gaussian component by using the minimum mean square error (MMSE) method as shown in Eq. 6.18,

$$\begin{aligned} \hat{\overline{x}}_i &= \mathbb{E}[\overline{x}_i | \overline{y}_i] \\ &= \overline{\mu}_{H_{\hat{j}_i}} + \boldsymbol{S}_{HL_{\hat{j}_i}} \boldsymbol{S}_{L_{\hat{j}_i}}^{-1} \left(\overline{y}_i - \overline{\mu}_{L_{\hat{j}_i}} \right) \end{aligned} \tag{6.18}$$

We find all the HR vectors $\{\overline{x}_i\}_{i=1}^{N}$, where $\overline{x}_i \in \mathbb{R}^{\tau^2 q^2}$, for the corresponding LR patches $\{\overline{y}_i\}_{i=1}^{N}$, where $\overline{y}_i \in \mathbb{R}^{\tau^2}$. Converting these HR vectors in patches and placing them on the HR grid, and averaging on the overlapping region produce the final HR image $\hat{X}$.

Algorithm 8 shows the process of obtaining the HR image from an LR image step-by-step.

6.3.3 Direct Approach versus Hierarchical Approach

In direct approach, we train GMM model using appropriate HR-LR patch sizes, e.g. for upsampling factor $\times 8$, we train using 32×32 and 4×4 patch sizes from HR and LR images of size 800×800 and 100×100. However, in Hierarchical approach, we train GMM model only for upsampling factor $\times 2$, but with different HR and LR images sizes, e.g. we train the first GMM model (called TrainGMM1) for factor $\times 2$ using

Algorithm 8 Multivariate GMM testing

```
**Input**: LR test image with extracted patch vectors $\{\overline{y}_i\}_{i=1}^{N}$, and the learnt GMM parameters $\{\omega_j, \overline{\mu}_j, \boldsymbol{S}_j\}_{j=1}^{K}$.
**Select Gaussian Component**: For a given test vector $\overline{y}_i$, find the Gaussian component (say $j^{th}$ component) which probably have generated it,
    $\hat{j}_i = \arg\max_{j \in 1 \cdots K} \gamma_{L_j}^{i}$
**Select HR vector**: From the probable $j^{th}$ Gaussian component, estimate the HR vector as,
    $\hat{\overline{x}}_i = \overline{\mu}_{H_{\hat{j}_i}} + \boldsymbol{S}_{HL_{\hat{j}_i}} \boldsymbol{S}_{L_{\hat{j}_i}}^{-1} \left( \overline{y}_i - \overline{\mu}_{L_{\hat{j}_i}} \right)$
**Post Processing**: On the overlapping regions on the HR image grid, a Gaussian weighted average is computed among the overlapping patches.
**Output**: The HR image $\hat{\boldsymbol{X}}$.
```

patch sizes 8×8 and 4×4 from HR and LR images of size 800×800 and 400×400 respectively; similarly, we train second GMM model (TrainGMM2) which is also for upsampling factor $\times 2$, but with smaller HR and LR image sizes of 400×400 and 200×200 respectively; and a third GMM model (TrainGMM3) for same upsampling factor and with same HR-LR patch sizes, but with different image sizes 200×200 and 100×100 for HR and LR training images.

Hence in hierarchical approach, we learn the small structure at the lowest resolution which will support to give accurate output at further stages for higher upsampling factors.

6.4 EXPERIMENTAL RESULTS AND DISCUSSIONS

In this section we will discuss on the results obtained by the SRGMM method and its comparison with other SR methods. Before looking at the results, we would like to present the database used for training and testing, and the LR image modeling used in the proposed SRGMM method.

6.4.1 Training/Testing Database

For GMM training, we have taken synthetic depth images from Mac Aodha et al. (2012) which the authors have used for training purpose for the problem of single depth image SR. There are total 31 synthetic depth examples, each of dimension 800×800, and all the examples were used for GMM training. Some of them are shown in Figure 6.1. A point to note here is that, the training images have darker pixel values for closer objects, and brighter pixel values for farthest object (which is based on the distance of object from the camera), whereas, the testing images which is taken from Middlebury database Scharstein and Szeliski (2002), have brighter values for near objects and the darker values for farthest object (which is based on the parallax distance of object from the two camera viewpoint).

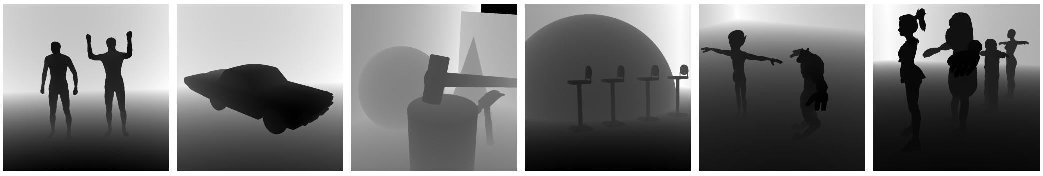


Figure 6.1: Few examples of synthetic training images

For testing purpose, we have used depth images from Middlebury dataset Scharstein and Szeliski (2002). We have chosen depth images which are having planar structures, and some images with varying number of objects from simple objects to complex objects. We experimented with adding noise to the training images to learn the LR-HR mapping better. The addition of noise to the training dataset was done with the intension of replicate the real scenario where noise is always implicit to the image and to avoid the mismatch between the test image and the training image. But we could not succeed in getting better results.

6.4.2 Parameter Selection

As mentioned earlier, we take the overlapped patches from both HR and LR training images, and vectorize them to concatenate to form a training matrix. We select N vectors ($N = 1,000,000$) randomly from the set of vectors $\overline{\boldsymbol{V}} = \{\overline{v}_i\}_{i=1}^{N}$ extracted from HR and LR training images. One can choose more number of vectors, which can further ease the possibility of finding the more closer match for the input test patch. Table 6.1 shows different HR and LR patches collected for different upsampling factor. We have experimented with various Gaussian mixtures (i.e. 50, 100, 150, 200, 250 and 300) to see how many Gaussian mixtures are suitable for the SR task.

Table 6.1: Selection of patch sizes for different upsampling factor

Upsampling factor	HR patch size	LR patch size
Factor ×2	8×8	4×4
Factor ×4	16×16	4×4
Factor ×8	24×24	3×3

6.4.3 SR Results on Synthetic Depth Images

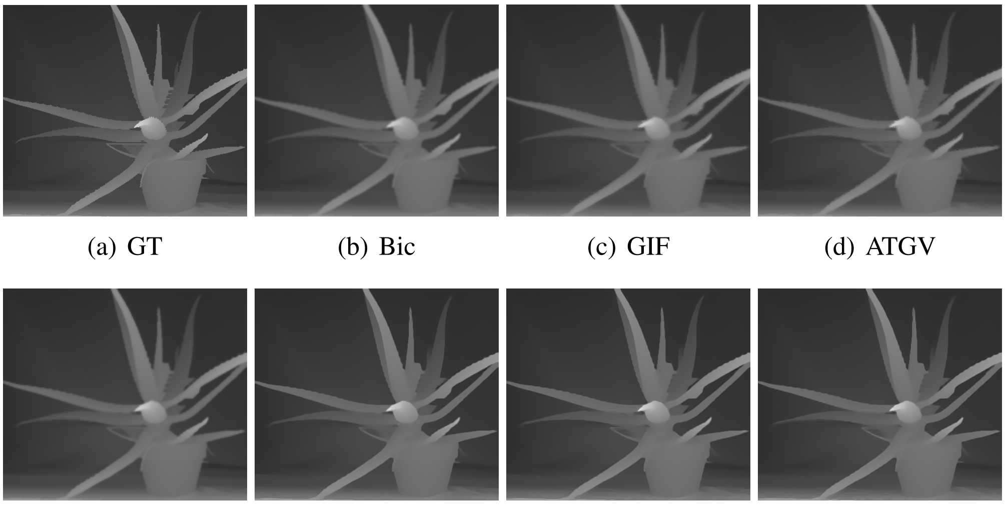
In this section we demonstrate the qualitative and quantitative results of depth image super-resolution methods on *noiseless* and *noisy* depth images for SR upsampling factors $\times 2$, $\times 4$ and $\times 8$. We compare SRGMM results with classical bilinear and bicubic interpolation results, and with SR results of other state-of-the-art methods like guided image filtering (GIF) He et al. (2010), anisotropic total generalized variation (ATGV) Ferstl et al. (2013), and residual interpolation (RI) Konno et al. (2015).

As we train the GMM model which has various parameters to set, we have experimented with different parameter values. We presented results for different number of Gaussian mixtures varying from 50 to 300 with a difference of 50. We have also demonstrated the effect of selection of HR-LR patch sizes as 8×8 and 4×4 for HR and LR patch respectively in one setting to [6,3] and [4,2] in another settings. We have also presented the results with different number of patch collections as 10 Lakhs, 5 Lakhs, 2.5 Lakhs and 1 Lakhs. The synthetic images Mac Aodha et al. (2012) were used from training GMM model, and we have used the standard Middlebury depth dataset Scharstein and Szeliski (2002) for testing the model. For testing purpose, we have shown the results variation with change in the target size of the SR image, hence we have presented results for different target images size as *one-third*, *one-half* and *full* sizes. Other than noiseless test inputs, we have also considered noisy test inputs to demonstrate the effectiveness of SRGMM method in suppressing the noise level in the SR output.

Qualitative Results: Figure 6.2 and Figure 6.3 shows the SR results for upsampling factor $\times 2$ on *aloe* depth image *without noise* ($\sigma = 0$) and *with noise* ($\sigma = 5$) respectively. We have shown SRGMM results obtained from three trained GMM models with 100, 200 and 300 Gaussian mixtures, and compare it with the SR results of GIF, ATGV and RI methods.

As observed in Figure 6.2, the outputs produced by SRGMM method with different are better in terms of preserving the edges, and maintaining the overall structure of the objects in the image. Compared to the other competitive SR methods, SRGMM results are more clear and distinct at the edge discontinuities. The bicubic results suffer severely by blurring the edge discontinuities, and the GIF, ATGV and RI methods also has some artifacts at the edges of the objects at different depths. SRGMM method is able to perform well with better smoothing at the object regions where the depth seems to be almost similar at the object surface, and is also good at preserving edge discontinuities with higher accuracy. We also show the results on noisy inputs in Figure 6.3, where we have added external noise to a clean test image with noise standard deviation of 5 ($\sigma = 5$). The interpolation results heavily suffer from noise and it is unable to suppress the noise level in the output image. Other SR methods like GIF, ATGV and RI methods perform quite well in terms of suppressing the noise level as these methods were proposed for noisy images. However, SRGMM method is better then those in terms of suppressing the noise and in terms of preserving the edge discontinuities. We have shown the SRGMM results under three different setup which considers the number of Gaussian mixtures as 100, 200 and 300 respectively. One can notice (more clearly in the SR results for higher upsampling factor) that as we consider more number of Gaussian mixtures the chances that the test patch can find its best close match increases, thereby producing better results.

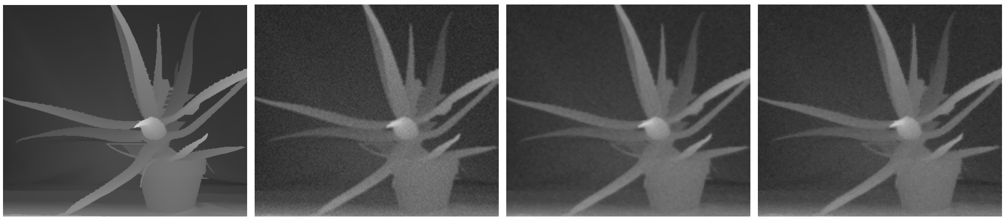
To show the strength of the proposed SRGMM method, we demonstrate it for higher upsampling factors. Figure 6.4 shows the SR results for upsampling factor ×4 on *noiseless* images of *cones*, and Figure 6.5 shows the SR results for upsampling factor ×8 on *noiseless* images of *teddy*. In Figure 6.4, we can see that the *vertical sticks* in the images are see clearly in output produced by SRGMM with much clear demarcation of stick and its background, and the overall depth information is also maintained. This information is not very clear (blurred) in either the bicubic interpolation output or the other SR methods and they suffer from blurring artifacts. As we see minutely (under high zoom), the sticks, the head, and the cones in the front have improved edge information and the object surface looks much smoother, whereas the ATGV and RI methods have some leaky artifacts at the edges of the objects at different depths. Similarly, Figure 6.5 shows the SR results but for the factor ×8 on noiseless images. Here, SRGMM method performs better than the classical interpolation methods, and we also perform better than the GIF method. The ATGV and RI methods takes a lead and perform better than SRGMM method in this scenario, because of the reason that the LR test patch will become so small for ×8 case that it becomes difficult to find a good HR match.


(a) GT (b) Bic (c) GIF (d) ATGV

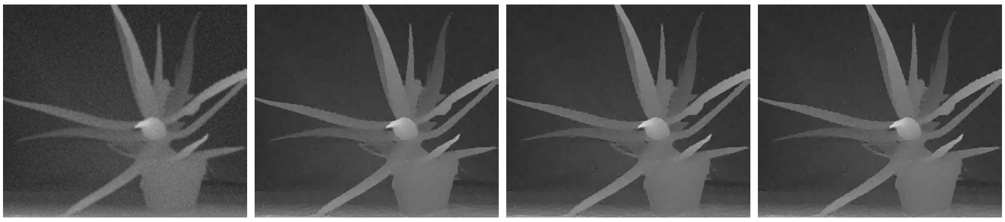
(e) RI (f) SRGMM 100 mix (g) SRGMM 200 mix (h) SRGMM 300 mix

Figure 6.2: Visual comparison of SR results upsampled by ×2 factor on noiseless image.

We also show the quantitative results on some of the selected test images from Middlebury dataset Scharstein and Szeliski (2002) in terms of PSNR and SSIM performance metrics. We have tabulated the SR results in Table 6.2, 6.3 and 6.4 for upsampling fac-

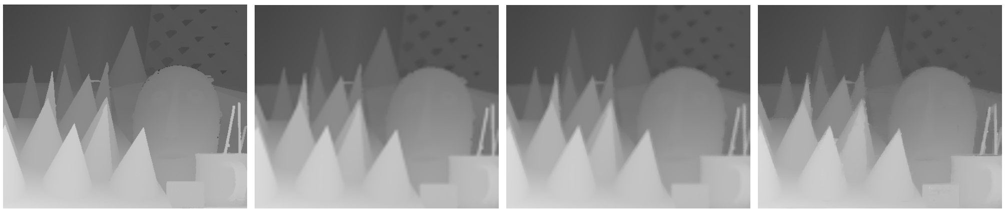


(a) GT (b) Bic (c) GIF (d) ATGV

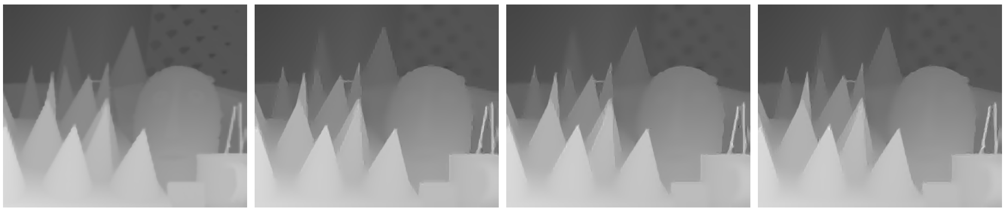


(e) RI (f) SRGMM 100 mix (g) SRGMM 200 mix (h) SRGMM 300 mix

Figure 6.3: Visual comparison of SR results upsampled by $\times 2$ factor on noisy image.



(a) GT (b) Bic (c) GIF (d) ATGV

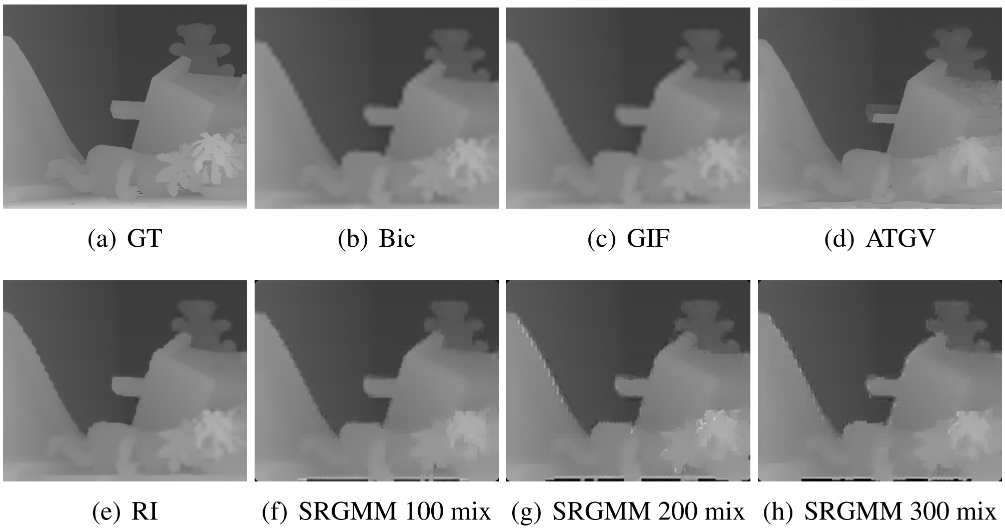


(e) RI (f) SRGMM 100 mix (g) SRGMM 200 mix (h) SRGMM 300 mix

Figure 6.4: Visual comparison of SR results upsampled by $\times 4$ factor on noiseless image.

tor $\times 2$, $\times 4$ and $\times 8$ respectively. Each table shows the SR results on the common set of test images *without noise* ($\sigma = 0$) and *with noise* ($\sigma = 5$), and compared SRGMM method trained over 100, 200 and 300 Gaussian mixtures with classical bilinear and bicubic interpolation methods and other state-of-the-art SR methods like GIF, ATGV and RI methods.

Quantitative Results: Table 6.2 shows PSNR/SSIM results for SR factor $\times 2$. In tables, a row with **bold** numbers represents the best result among all the comparative



(a) GT (b) Bic (c) GIF (d) ATGV

(e) RI (f) SRGMM 100 mix (g) SRGMM 200 mix (h) SRGMM 300 mix

Figure 6.5: Visual comparison of SR results upsampled by $\times 8$ factor on noiseless image.

methods. We can observe that the proposed SRGMM method (last 3 columns) shows best performance among the comparative methods. In *noisless* case, the GMM trained with 200 Gaussian mixtures performs best on most of the test images, and even otherwise one of the variants of proposed SRGMM method with either 100, 200 or 300 produce better results as compared to other single depth image SR methods like GIF, ATGV and RI. Same is the case with *noisy* scenario, but the GMM trained with 200 and 300 Gaussian mixtures perform equally well. Estimating the optimal number of Gaussian mixtures required in this case is difficult as it depends on many different parameters. There is no proper value which can give consistent results over the set of test images, however one can freely choose the number of Gaussian mixtures between 300 to 500 for depth image super-resolution problem.

Table 6.3 and Table 6.4 shows SR results for upsampling factor 4 and 8. The GMM trained with 300 Gaussian mixture for SR factor $\times 4$ does better job of giving the best results among the comparative methods, but the SRGMM method with different Gaussian mixtures does not perform well for all the images. This can be seen more in *noisy* case were either ATGV or RI method lead the track. And as we see SR results for factor $\times 8$, the SRGMM method lags behind, and RI method performs better in both *noiseless* and *noisy* case.

Table 6.2: PSNR/SSIM performance metrics for SRGMM for upsampling factor ×2 (**Bold** represents the best result among comparative methods)

Images	Bil	Bic	GIF	ATGV	RI	SRGMM 100 Mix	SRGMM 200 Mix	SRGMM 300 Mix
	x2_sig0							
Aloe	33.36/0.95	33.67/0.95	33.83/0.96	34.41/0.96	35.42/0.97	37.96/0.98	37.76/0.98	**37.98/0.98**
Art	31.02/0.92	31.42/0.93	31.70/0.93	32.01/0.94	32.76/0.95	35.52/0.97	**35.88/0.97**	35.64/0.97
Baby	37.95/0.98	38.27/0.98	38.44/0.98	39.16/0.98	40.15/0.99	43.70/0.99	**43.95/0.99**	43.84/0.99
Books	39.15/0.98	39.47/0.98	39.63/0.98	40.11/0.98	41.14/0.98	43.18/0.99	**43.88/0.99**	43.51/0.99
Cones	36.25/0.97	36.58/0.97	36.75/0.97	37.50/0.97	38.09/0.98	40.00/0.98	**40.00/0.98**	39.98/0.98
Moebius	39.67/0.97	40.01/0.98	40.22/0.98	40.82/0.98	41.51/0.98	**42.26/0.98**	42.12/0.98	42.23/0.98
Plastic	39.28/0.99	39.57/0.99	40.14/0.99	41.64/0.99	41.72/0.99	45.76/0.99	**46.15/1.00**	46.07/1.00
Reindeer	34.17/0.96	34.51/0.97	34.87/0.97	35.10/0.97	35.99/0.98	38.64/0.98	38.88/0.98	**38.95/0.98**
Sawtooth	37.50/0.98	37.86/0.98	38.71/0.98	38.76/0.98	39.51/0.99	44.13/1.00	**44.73/1.00**	44.58/1.00
Teddy	38.83/0.98	39.15/0.98	39.52/0.98	40.13/0.98	40.70/0.98	41.21/0.98	39.82/0.98	**41.60/0.98**
Venus	42.35/0.99	42.69/0.99	43.25/0.99	43.87/0.99	44.48/0.99	48.90/1.00	**49.29/1.00**	48.97/1.00
	x2_sig5							
Aloe	32.15/0.87	31.70/0.81	33.12/0.92	32.56/0.85	32.51/0.82	**35.60/0.96**	35.40/0.96	35.46/0.96
Art	30.27/0.84	30.12/0.79	31.24/0.90	30.96/0.85	30.86/0.80	33.70/0.95	32.83/0.95	**33.84/0.95**
Baby	35.05/0.88	33.94/0.82	36.68/0.94	35.54/0.88	34.38/0.82	36.69/0.93	**36.76/0.93**	36.63/0.92
Books	35.64/0.88	34.37/0.82	37.55/0.94	35.92/0.88	34.70/0.82	39.79/0.97	39.77/0.98	**40.22/0.98**
Cones	34.16/0.88	33.33/0.82	35.54/0.94	34.56/0.87	33.83/0.83	37.94/0.97	**37.94/0.97**	37.76/0.97
Moebius	35.84/0.88	34.51/0.82	37.72/0.94	36.13/0.88	34.76/0.82	38.98/0.96	**39.02/0.96**	38.96/0.96
Plastic	35.75/0.89	34.44/0.82	38.00/0.95	37.01/0.89	34.85/0.82	41.77/0.98	**41.90/0.98**	41.71/0.98
Reindeer	32.76/0.87	32.21/0.81	34.10/0.94	33.77/0.88	32.86/0.82	36.59/0.97	36.70/0.97	**36.71/0.97**
Sawtooth	34.81/0.88	33.78/0.81	36.96/0.94	35.15/0.87	34.17/0.81	37.35/0.94	**37.61/0.94**	37.60/0.94
Teddy	35.56/0.89	34.34/0.83	37.49/0.94	35.82/0.88	34.66/0.83	38.58/0.97	37.51/0.97	**38.74/0.97**
Venus	36.77/0.88	35.11/0.82	39.48/0.95	36.83/0.87	35.27/0.82	39.22/0.95	**39.48/0.95**	39.24/0.95

Table 6.3: PSNR/SSIM performance metrics for SRGMM for upsampling factor ×4 (**Bold** represents the best result among comparative methods)

Images	Bil	Bic	GIF	ATGV	RI	SRGMM 100 Mix	SRGMM 200 Mix	SRGMM 300 Mix
	x4_sig0							
Aloe	30.10/0.92	30.11/0.92	30.47/0.93	31.90/0.94	34.82/0.97	34.72/0.96	**34.84/0.96**	34.81/0.96
Art	28.26/0.88	28.44/0.88	28.87/0.89	29.80/0.91	31.61/0.94	31.83/0.93	32.35/0.94	**32.42/0.94**
Baby	34.78/0.97	34.87/0.97	35.19/0.97	36.89/0.98	39.25/0.99	39.36/0.98	39.80/0.98	**40.25/0.99**
Books	36.08/0.96	36.18/0.96	36.49/0.97	37.90/0.97	**40.67/0.98**	38.21/0.97	39.52/0.98	38.63/0.98
Cones	33.08/0.95	33.12/0.95	33.56/0.95	35.31/0.96	36.83/0.97	36.69/0.97	36.71/0.97	**36.91/0.97**
Moebius	36.43/0.96	36.53/0.96	36.89/0.96	38.20/0.97	**40.87/0.98**	38.18/0.96	38.82/0.97	38.78/0.97
Plastic	36.27/0.98	36.33/0.98	36.76/0.98	**41.55/0.99**	38.47/0.99	38.65/0.98	40.97/0.99	39.66/0.98
Reindeer	31.16/0.94	31.25/0.94	31.73/0.95	34.01/0.97	35.12/0.97	35.09/0.96	35.40/0.97	**35.54/0.97**
Sawtooth	34.84/0.97	35.04/0.97	35.80/0.97	37.61/0.98	38.76/0.99	40.24/0.99	40.40/0.99	**40.46/0.99**
Teddy	35.67/0.97	35.78/0.97	36.26/0.97	38.30/0.98	**39.78/0.98**	37.97/0.97	38.00/0.97	37.90/0.97
Venus	39.63/0.98	39.81/0.98	40.43/0.99	42.62/0.99	44.54/0.99	44.23/0.99	44.99/0.99	**45.48/0.99**
	x4_sig5							
Aloe	29.39/0.87	29.08/0.84	29.76/0.88	31.24/0.91	32.25/0.88	33.06/0.93	32.91/0.93	**32.96/0.93**
Art	27.79/0.83	27.72/0.80	28.36/0.85	29.41/0.89	30.19/0.85	30.99/0.91	**31.18/0.91**	31.05/0.91
Baby	33.08/0.90	32.41/0.87	33.54/0.92	**35.77/0.95**	34.36/0.89	34.55/0.91	34.60/0.90	34.42/0.90
Books	33.85/0.90	33.06/0.87	34.39/0.92	36.21/0.94	34.71/0.89	36.68/0.96	**37.06/0.96**	36.68/0.96
Cones	31.90/0.89	31.39/0.86	32.44/0.91	34.31/0.93	33.39/0.88	**35.26/0.95**	35.14/0.95	35.16/0.95
Moebius	34.07/0.90	33.24/0.86	34.51/0.91	36.64/0.94	34.79/0.88	**36.51/0.95**	36.43/0.95	36.25/0.94
Plastic	34.01/0.91	33.18/0.87	34.63/0.93	**39.20/0.97**	34.55/0.89	37.95/0.97	38.21/0.97	37.72/0.97
Reindeer	30.31/0.88	30.00/0.85	30.91/0.91	33.22/0.94	32.44/0.88	33.83/0.95	**33.85/0.94**	33.70/0.94
Sawtooth	33.03/0.90	32.43/0.86	33.92/0.92	**35.98/0.95**	33.99/0.88	35.50/0.93	35.27/0.93	35.02/0.92
Teddy	33.62/0.91	32.90/0.87	34.24/0.93	**36.50/0.95**	34.52/0.89	35.91/0.95	35.68/0.95	35.48/0.95
Venus	35.57/0.91	34.43/0.87	36.42/0.94	**39.14/0.95**	35.39/0.89	37.14/0.94	36.98/0.93	36.54/0.93

Table 6.4: PSNR/SSIM performance metrics for SRGMM for upsampling factor ×8 (**Bold** represents the best result among comparative methods)

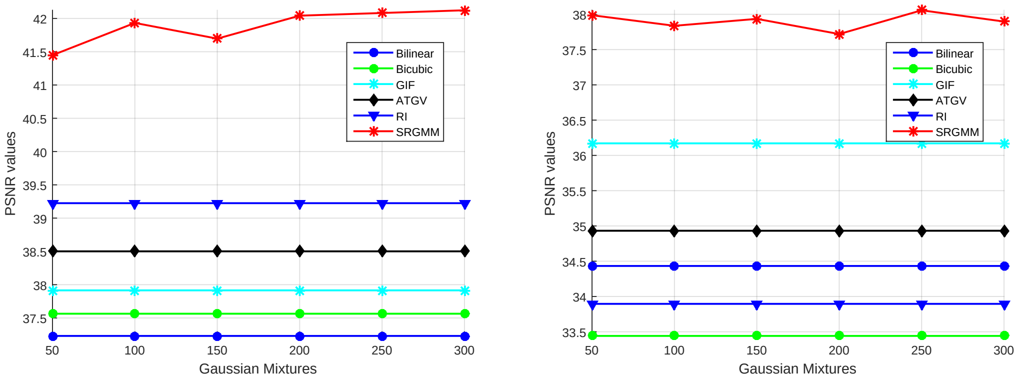
Images	Bil	Bic	GIF	ATGV	RI	SRGMM 100 Mix	SRGMM 200 Mix	SRGMM 300 Mix
				x8_sig0				
Aloe	26.52/0.88	26.22/0.87	26.36/0.88	26.12/0.90	**30.27/0.92**	29.54/0.90	29.10/0.89	29.40/0.89
Art	24.90/0.82	24.77/0.81	24.91/0.82	25.95/0.86	**28.05/0.88**	26.05/0.83	24.37/0.82	23.90/0.82
Baby	30.56/0.94	30.37/0.94	30.67/0.94	32.52/0.96	**36.35/0.97**	33.14/0.95	32.60/0.95	32.64/0.95
Books	31.76/0.95	31.58/0.94	31.83/0.95	32.69/0.94	**36.96/0.97**	32.51/0.95	32.34/0.95	31.51/0.94
Cones	29.79/0.92	29.59/0.92	29.79/0.93	30.72/0.93	**32.63/0.95**	30.45/0.92	30.10/0.92	30.42/0.92
Moebius	32.71/0.94	32.60/0.94	32.72/0.94	32.80/0.94	**37.19/0.96**	29.42/0.93	26.66/0.93	26.24/0.93
Plastic	31.11/0.96	30.93/0.96	31.08/0.96	33.46/0.97	**36.39/0.98**	33.31/0.97	33.43/0.96	32.45/0.96
Reindeer	27.84/0.91	27.66/0.91	28.02/0.92	29.64/0.94	**31.39/0.95**	28.94/0.92	28.23/0.90	28.21/0.90
Sawtooth	31.16/0.95	31.11/0.95	31.58/0.95	32.21/0.96	**34.23/0.97**	32.85/0.96	32.63/0.95	32.65/0.95
Teddy	31.70/0.95	31.58/0.94	31.83/0.95	33.09/0.96	**35.90/0.97**	27.75/0.94	24.94/0.94	25.23/0.94
Venus	36.08/0.97	35.99/0.97	36.36/0.97	38.61/0.97	**41.11/0.99**	36.96/0.98	36.71/0.97	36.69/0.97
				x8_sig5				
Aloe	26.20/0.86	25.79/0.84	25.98/0.85	25.87/0.89	**29.40/0.89**	28.98/0.88	28.42/0.87	28.33/0.86
Art	24.68/0.80	24.45/0.78	24.62/0.79	25.87/0.86	**27.25/0.84**	25.69/0.82	23.13/0.80	22.90/0.80
Baby	29.77/0.92	29.29/0.90	29.69/0.91	32.31/0.96	**32.62/0.93**	31.43/0.92	31.00/0.91	30.99/0.91
Books	30.82/0.92	30.28/0.91	30.67/0.92	32.84/0.95	**33.35/0.93**	30.66/0.93	28.99/0.93	29.09/0.92
Cones	29.24/0.90	28.82/0.89	29.12/0.90	30.45/0.93	**31.12/0.91**	30.16/0.91	29.33/0.90	29.84/0.90
Moebius	31.66/0.92	31.09/0.90	31.41/0.91	32.57/0.94	**33.63/0.92**	28.71/0.92	26.23/0.91	26.07/0.90
Plastic	30.28/0.94	29.77/0.92	30.07/0.93	**33.81/0.97**	32.97/0.94	32.27/0.95	32.26/0.95	32.55/0.94
Reindeer	27.44/0.89	27.11/0.88	27.52/0.89	29.56/0.94	**30.00/0.91**	28.43/0.90	28.06/0.89	27.79/0.88
Sawtooth	30.22/0.92	29.80/0.91	30.46/0.92	**32.56/0.96**	31.87/0.93	31.37/0.93	30.49/0.91	30.60/0.91
Teddy	30.76/0.92	30.28/0.91	30.66/0.92	32.85/0.96	**33.05/0.93**	27.72/0.92	27.06/0.91	24.81/0.91
Venus	33.82/0.94	32.98/0.93	33.54/0.94	**38.67/0.98**	34.71/0.94	33.76/0.94	32.81/0.93	32.04/0.93

To show graphically, we have computed the average PSNR value over the chosen test set and we plot it against the number of Gaussian mixtures used for GMM training. Since other SR methods and interpolation methods are independent of the number of Gaussian mixtures, so we represent their average plot as straight line (constant). We show the trend of SRGMM method (red curve) for upsampling factor ×2, ×4 and ×8 on noiseless and noisy images in Figure 6.6 and their averaged PSNR values in Table 6.5. The plots show clearly how many number of Gaussian mixtures are sufficient for depth image super-resolution.

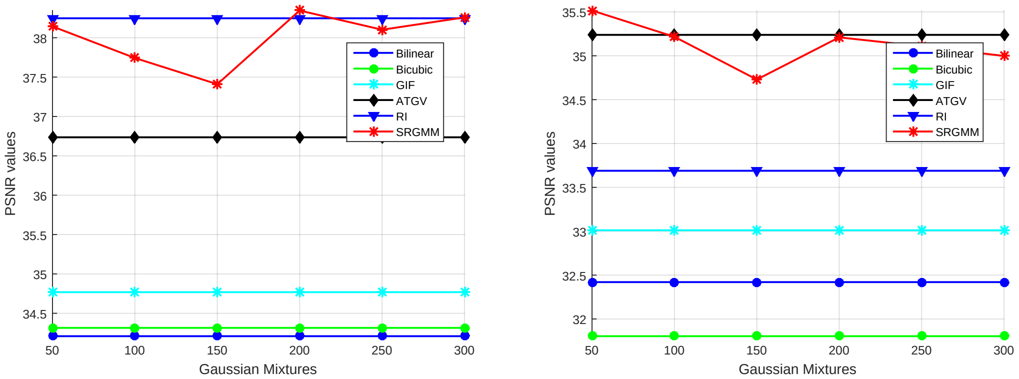
Table 6.5: Average PSNRs of SRGMM method with different Gaussian Mixtures for single depth image SR problem (**Bold** represents the best result among comparative methods)

Upsampling	Bil	Bic	GIF	ATGV	RI	SRGMM 50 Mix	SRGMM 100 Mix	SRGMM 150 Mix	SRGMM 200 Mix	SRGMM 250 Mix	SRGMM 300 Mix
x2_Sig0	37.23	37.56	37.91	38.50	39.22	41.44	41.93	41.69	42.04	42.08	**42.12**
x2_Sig5	34.43	33.44	36.17	34.93	33.89	37.98	37.83	37.93	37.72	**38.05**	37.89
x4_Sig0	34.20	34.31	34.76	36.73	38.24	38.14	37.74	37.41	**38.34**	38.10	38.25
x4_Sig5	32.42	31.80	33.01	35.23	33.68	**35.51**	35.21	34.72	35.21	35.10	34.99
x8_Sig0	30.37	30.21	30.46	31.61	**34.58**	31.61	30.99	29.63	30.10	30.26	29.94
x8_Sig5	29.53	29.06	29.43	31.57	**31.81**	30.22	29.92	29.09	28.88	29.05	28.63

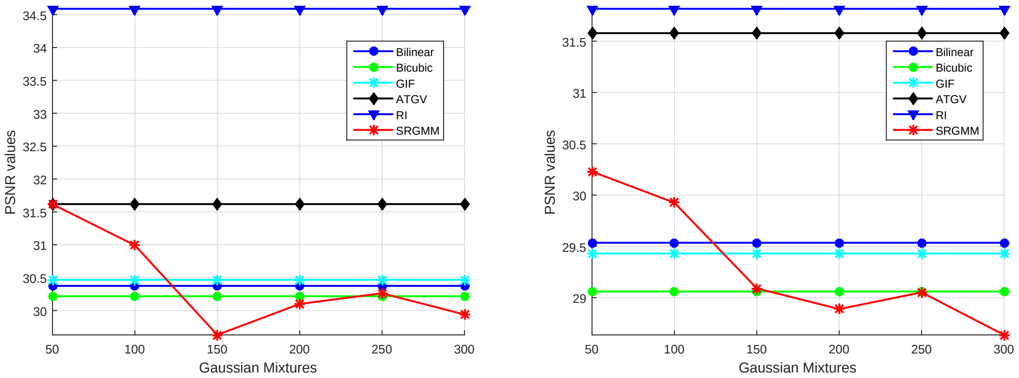
We can see that for upsampling factor ×2, the SRGMM performs better than the



(a) x2_Sig0 (b) x2_Sig5



(c) x4_Sig0 (d) x4_Sig5



(e) x8_Sig0 (f) x8_Sig5

Figure 6.6: Comparison of average PSNR values of depth super-resolution methods with different number of Gaussian mixtures for upsampling factors ×2, ×4 and ×8 on *noiseless* and *noisy* images.

interpolation results and also does well as compared to other SR methods in both noiseless and noisy cases. For upsampling factor ×4, SRGMM method performs better than all other SR methods at 200 Gaussian mixtures for noiseless images, and at 50 Gaussian mixtures for noisy images. For upsampling factor $\times 8$, for both noiseless and noisy images, SRGMM method does better than the classical interpolation methods at 50 and 100 Gaussian mixtures, but it does not perform well when compared to other SR methods like GIF, ATGV and RI methods. The reason could be that, the methods like GIF, ATGV and RI make use of the guidance image, and the guidance image is a strong prior to get good results for higher upsampling factor. The quality of the output produced by the SRGMM method is purely based on the best close match found for test patch from the trained models. However, during testing for higher upsampling factor, the patches of test images will have very few information in it to find the proper match from the the trained HR-LR patch pair. This is the reason why trained SRGMM models works well till upsampling factor $\times 4$ when target resolution is of *one-third* size (nearly 420×365), and it gives poor results for higher upsampling factor ($\times 8$ or more).

We would like to note here that, the ATGV and RI method are mainly proposed for noisy cases and uses *full* resolution images as target image for their SR reconstruction. To prove a point that the target resolution also plays important role in producing better results, we experimented with different target resolution for all the SR factors and compare the results.

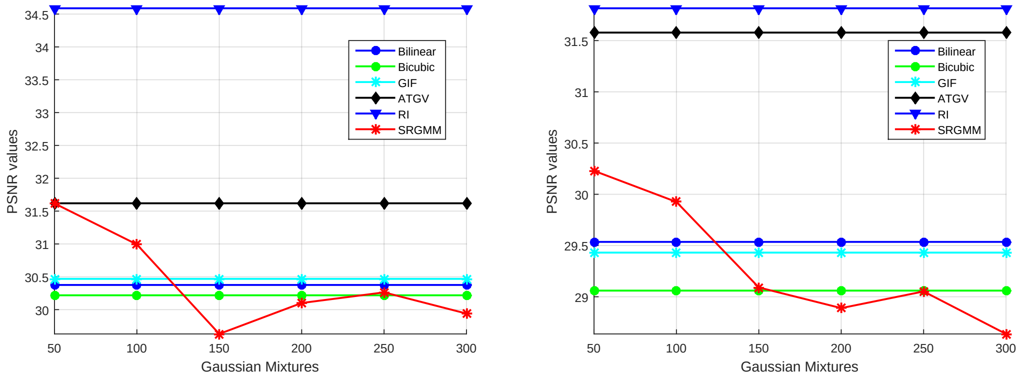
With a limitations on target output dimension to be somewhere close to 420×365, the input LR image will be of size somewhere close to 52×45 for SR factor $\times 8$. Such a small icon size image will not have much information (in terms of edges or corners) to retrieve a proper match from the HR-LR patch pair learnt during training. Thus, to see the effect of target spatial resolution on the output performance produced by the training models, we have experimented with the *full* size image set for testing (nearly 1260×1100), for which the LR image for upsampling factor $\times 8$ would be of size close to 160×140. We have noticed that, in case of *full* size images, the LR image for upsampling factor $\times 8$ will have sufficient resolution of 160×140, so that the patches generated from it does have some meaningful information, and thus, on matching with the learnt HR-LR patches pairs it finds a suitable match for SR reconstruction.

We have tabulated the results only for $\times 8$ upsampling factor with *one-third* and *full* size images considering both *noiseless* and *noisy* scenarios. These results are shown in Table 6.6, and we have observed that, relatively, the SRGMM method gives better results when *full* size images are considered as the target resolution for upsampling. The plots of which is shown in Figure 6.7, where we again compare SRGMM method (red curve) with other SR methods over different number of Gaussian mixtures. In Figure 6.7, the top row shows graphical results of SR by factor 8 on *one-third* size images, and the bottom row shows the plot of SR results for factor 8 but on *full* size images. For *noiseless* scenario with *one-third* size images, the SRGMM method produces good results compared to ATGV and GIF methods, but could not perform better than RI method. The reason could be that, the RI method is a guidance image based techniques, and this prior is a stronger prior for SR problem to address for higher upsampling factors. However, we notice that, on its counterpart for *noisy* scenario the SRGMM method with 50 and 100 Gaussian mixtures performs marginally better than GIF method. However, if we choose to have the target resolution as *full* size images, then in *noiseless* case the SRGMM method performs better ($>$ 1dB on average) with 50, 100 and 150 Gaussian mixtures, and performs marginally better with 200, 250 and 300 Gaussian mixtures, but could not perform better than RI method. On counterpart, in *noisy* case, we could perform better than RI method over 50, 100 and 150 Gaussian mixtures. In all the cases, the SRGMM method perform way better than the classical bilinear and bicubic interpolation method.

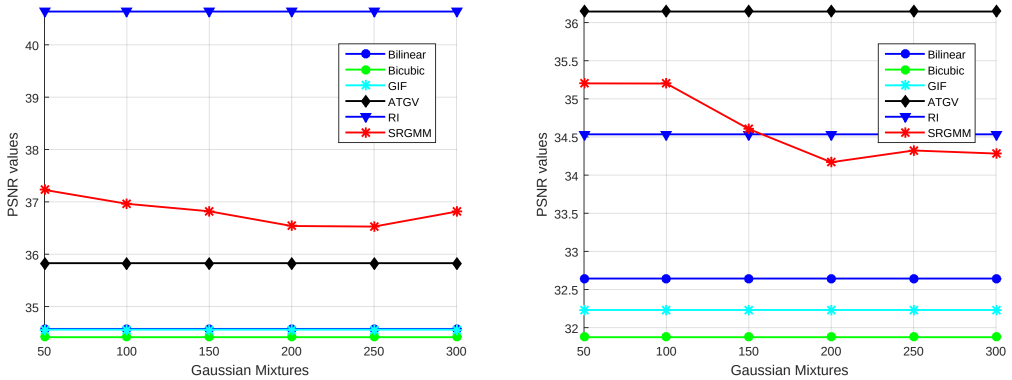
Table 6.6: Comparison of average PSNRs of SRGMM method with different Gaussian mixtures with other various depth image SR methods on one-third resolution and full resolution *noiseless* and *noisy* images (**Bold** represents the best result among comparative methods)

Upsampling	Bil	Bic	GIF	ATGV	RI	SRGMM 50 Mix	SRGMM 100 Mix	SRGMM 150 Mix	SRGMM 200 Mix	SRGMM 250 Mix	SRGMM 300 Mix
x8_Sig0_Third	30.37	30.21	30.46	31.61	**34.58**	31.61	30.99	29.63	30.10	30.26	29.94
x8_Sig0_Full	34.57	34.41	34.56	34.56	**40.63**	37.22	36.96	36.82	36.53	36.52	36.81
x8_Sig5_Third	29.53	29.06	29.43	31.57	**31.81**	30.22	29.92	29.09	28.88	29.05	28.63
x8_Sig5_Full	32.64	31.87	32.23	32.23	34.53	**35.20**	35.20	34.60	34.16	34.32	34.28

We have even demonstrated the effect of HR-LR patch sizes on the quality of the SR results. We have performed this experiment *only* for $\times 2$ upsampling factor. The first set of experiments are performed by training the GMM model with HR-LR patch sizes of 8×8 and 4×4 respectively, and the second experiment is with 6×6 and 3×3,



(a) x8_Sig0_Third (b) x8_Sig5_Third



(c) x8_Sig0_Full (d) x8_Sig5_Full

Figure 6.7: Plot of average PSNRs of SRGMM method with different Gaussian mixtures with other various depth image SR methods on *one-third* resolution and *full* resolution *noiseless* and *noisy* images.

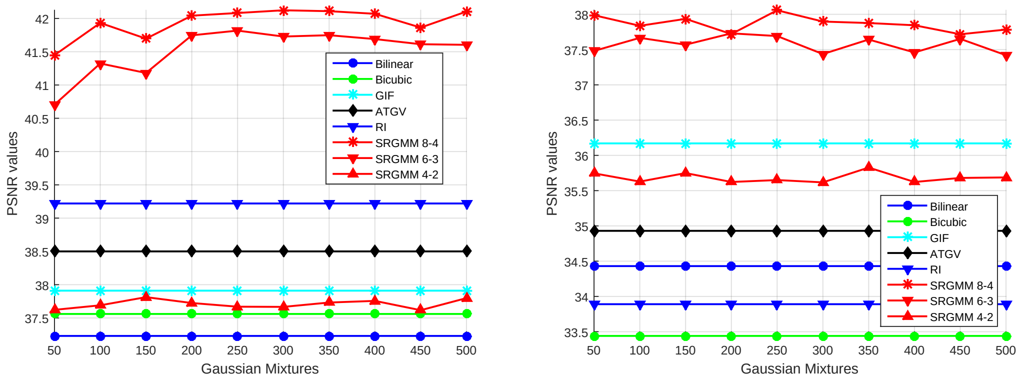
and third experiment is with 4×4 and 2×2. The average PSNR results are tabulated and shown in Table 6.7, and their graphical plot is shown in Figure 6.8. Same as earlier, we have computed the average PSNR values over the test set and plot them against the different number of Gaussian mixtures. The red curves are the plots of average PSNR values obtained from SRGMM method trained with different patch sizes on noiseless and noisy test images.

In Figure 6.8 we see that, the GMM training with smaller patch sizes *reduces* the overall results as compared to its training with bigger patch sizes. For both *noiseless* and *noisy* cases, the GMM trained with HR-LR patch sizes as 8×8 and 4×4 (represented as 8-4) gives better results when compared to GMM trained with 6-3 or 4-2 case. For noiseless case, the SRGMM method performs better with HR-LR patch sizes 8-4 as compared to 4-2, and we perform much better than other comparative methods with high margin. For 6-3 patch sizes also, SRGMM method method performs well as compared to all other methods, but as we reduce the patch sizes further to 4-2 the performance degrades heavily as shown in Figure 6.8, but still beat the bilinear and bucibic interpolation results. The reason for performance degradation is because, finding the HR patch for the corresponding input test LR patch from a selected Gaussian component is difficult as many LR patches from the Gaussian component will find a close match to the test LR patch, and hence a selected LR patch, with a small difference in its pixel values as compared to the actual value will leads to a large variation in its corresponding HR patch. The same trend is observed for noisy images also. With 8-4 and 6-3 patch sizes, SRGMM results are better than all other SR methods, but for 4-2 patch size we are not able to perform better than all the comparative methods, but still we better than ATGV and RI methods.

Table 6.7: Average PSNRs of SRGMM method for upsampling factor ×2, with different Gaussian Mixtures, for single depth image SR problem (**Bold** represents the best result among comparative methods)

Upsampling factor ×2	Bil	Bic	GIF	ATGV	RI	SRGMM 50 Mix	SRGMM 100 Mix	SRGMM 150 Mix	SRGMM 200 Mix	SRGMM 250 Mix	SRGMM 300 Mix	SRGMM 350 Mix	SRGMM 400 Mix	SRGMM 450 Mix	SRGMM 500 Mix
	Noiseless case														
HR 8x8 & LR 4x4	37.23	37.56	37.91	38.50	39.22	41.44	41.93	41.69	42.04	42.08	**42.12**	42.10	42.07	41.86	42.10
HR 6x6 & LR 3x3	37.23	37.56	37.91	38.50	39.22	40.70	41.32	41.18	41.74	**41.81**	41.72	41.74	41.69	41.61	41.60
HR 4x4 & LR 2x2	37.23	37.56	37.91	38.50	**39.22**	37.62	37.69	37.81	37.72	37.66	37.66	37.73	37.75	37.61	37.79
	Noisy case														
HR 8x8 & LR 4x4	34.43	33.44	36.17	34.93	33.89	37.98	37.83	37.93	37.72	**38.05**	37.89	37.87	37.84	37.71	37.78
HR 6x6 & LR 3x3	34.43	33.44	36.17	34.93	33.89	37.48	37.66	37.56	**37.73**	37.69	37.43	37.64	37.45	37.65	37.42
HR 4x4 & LR 2x2	34.43	33.44	**36.17**	34.93	33.89	35.74	35.62	35.74	35.62	35.65	35.61	35.82	35.62	35.68	35.68

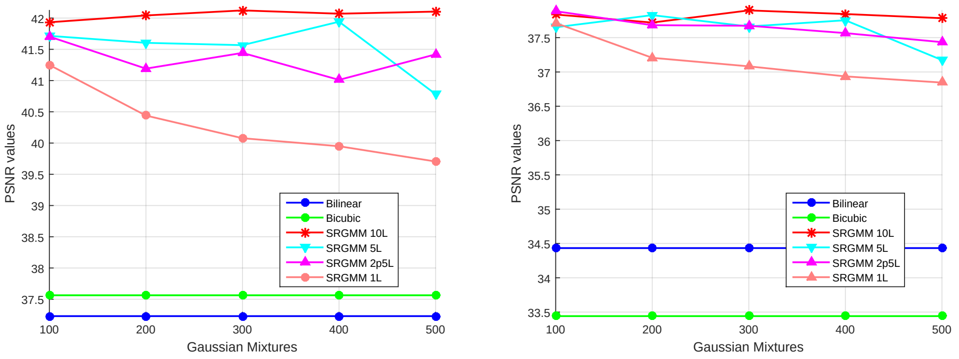
As depth image SR methods primary affair is to preserve the edges, we could see that most of the region is planar and have smooth or linearly smooth variations, hence we have experimented with lower number of patches also. Other than 10 lakh (10L), we also choose 5L, 2.5L (or 2p5L) and 1L number of HR-LR patches, by keeping the fixed HR-LR patch size to 8-4. We have seen nearly similar results over different experiments with different number of training patches, because depth images have mostly smooth regions with edge discontinuities, and we do not need many example patches for learning HR-LR mapping of smoother regions. For this scenario also, we have computed average PSNR values for different number of patches, whose plot is shown in



(a) x2_Sig0　　(b) x2_Sig5

Figure 6.8: Comparison of average PSNR values of depth super-resolution by factor ×2 with different HR-LR patch sizes.

Figure 6.9. Hence, we can say that less number of training patches are sufficient for training for depth image SR problems.



(a)　　(b)

Figure 6.9: SRGMM performance analysis with different number of training patches for upsampling factor ×2, with fixed HR-LR patch size (i.e. 8-4). (a) on noiseless images, (b) on noisy images.

6.4.4 SR Results From Direct and Hierarchical Approach

In this section we present the depth image SR results of the proposed SRGMM method and its comparison with other SR methods. Although we have trained GMM model with various Gaussian mixtures, but here we present the results only for 50 and 100 Gaussian mixtures, which were the best. Increasing the number of Gaussians further do not improve the results. We have chosen 8 test images from Middlebury dataset (Scharstein and Szeliski, 2002) and we compare the direct and hierarchical SRGMM results with bilinear and bicubic interpolation results and few other state-of-the-art depth image SR methods like ATGV (Ferstl et al., 2013), GIF (He et al., 2010) and RI (Konno et al., 2015) both qualitative and quantitative.

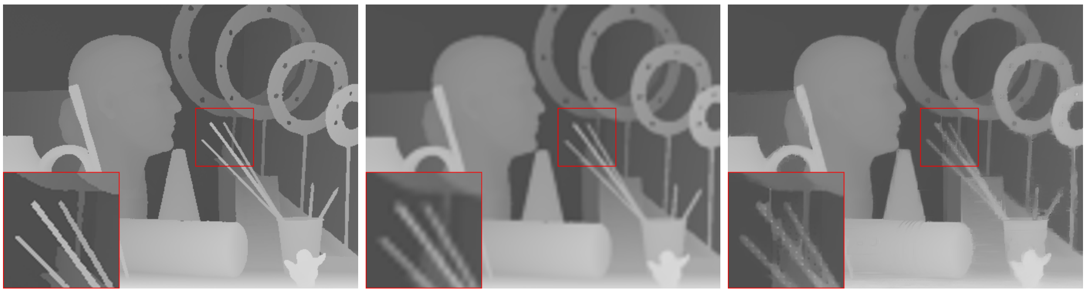
To simulate the LR data, we use the LR model shown in Eq. 1.3 earlier. For *noiseless* LR image creation the HR image X is only downsampled and blurred without having noise term ($\sigma = 0$). However, for *noisy* LR image creation the equation remains the same, but now with noise term in it with standard deviation $\sigma = 5$. We do not use this model anywhere in the proposed SRGMM method.

Noiseless scenario:

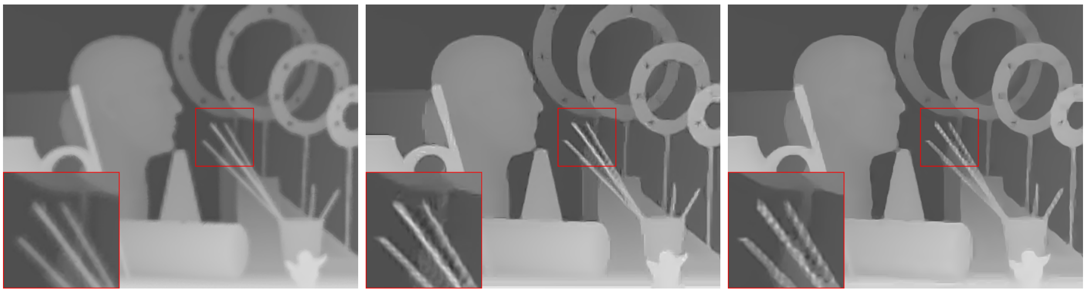
Figure 6.10 and Figure 6.11 shows the comparative results for SR by factor $\times 4$ and $\times 8$ of depth image *Art* and *Baby* respectively. As we can see that the SRGMM method (either direct or hierarchical) does better job in retaining the edge discontinuities, and the overall shape of the image is retained without any artifacts. On contrary, we observe that ATGV method is not been able to preserve edges to larger extent and exhibit some jagged artifacts around edges, and similar is the case with GIF method. As shown in the inline zoomed region of the portion of the image in Figure 6.10 (bottom row), the sticks in *Art* image produced by SRGMM method has clear distinction from the background and has sharp edge discontinuities. Similarly, the arms in *Baby* images in Figure 6.11 (bottom row) is sharper in SRGMM method with hierarchical approach than the interpolation methods and other state-of-the-art methods.

Table 6.8 shows the PSNR and SSIM (Wang et al., 2004) results on more images for *noiseless* scenario. We highlight the best score in *red* and second highest score in *blue* colour. We observe that the SRGMM method performs better than the classical interpolation methods for almost all the upsampling factors, and we also perform better than the state-of-the-art methods like ATGV, GIF and RI methods for most images. For upsampling factor $\times 2$, the SRGMM direct approach with 100 Gaussian mixtures performs better then all the other contemporary methods. For upsampling factor $\times 4$, the average results of the SRGMM direct approach with 50 Gaussian mixture is best.

Among the direct and hierarchical versions of the proposed SRGMM method, we note that at upsampling factor 4 the direct approach seems sufficient to learn the HR-LR mapping and yields better results than the hierarchical approach. However, for upsampling factor 8, clearly the hierarchical structure of learning GMM seems to help in producing better results, as the loss of information may be too high for a direct learning method.



(a) GT (b) Bic (c) ATGV (Ferstl et al., 2013)

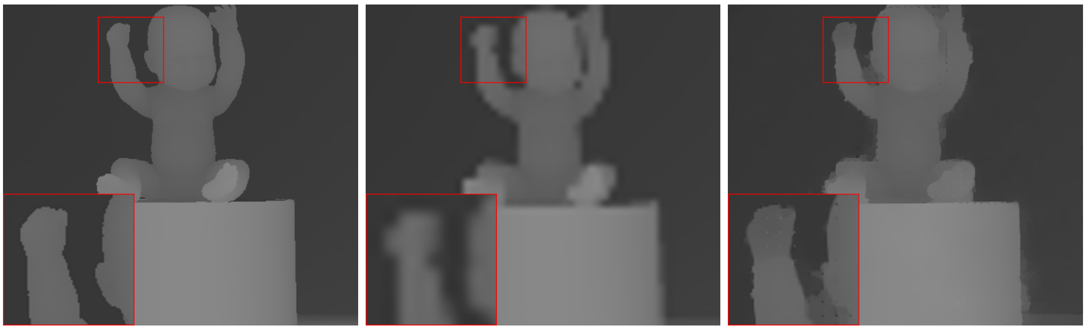


(d) GIF (He et al., 2010) (e) SRGMM-Dir 100Mix (f) SRGMM-Iter 100Mix

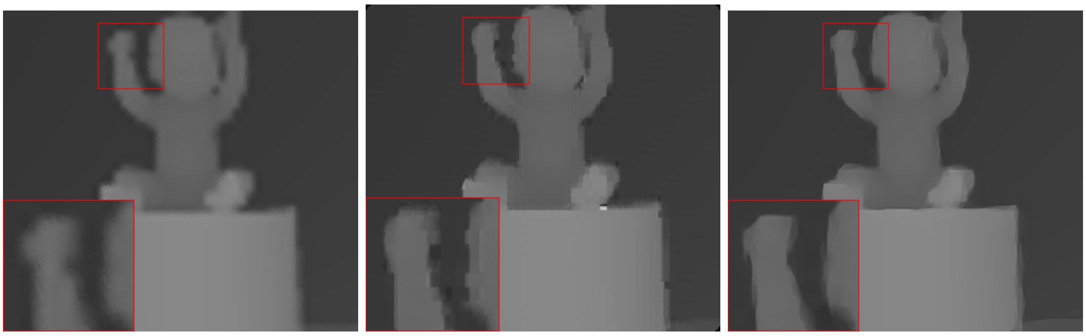
Figure 6.10: Qualitative results comparison for SR by factor $\times 4\_n0$ (Image: *Art*)

Noisy scenario: The GMM training for noisy scenario is similar to the the training procedure followed for noiseless scenario, but with the included noise term for LR image set generation. The HR image set is as usual the high-resolution images, but the LR image set is generated using Eq. 1.4 with noise term of standard deviation $\sigma = 5$.

Figure 6.12 and Figure 6.13 shows the testing results and the comparisons of both direct and hierarchical approach of SRGMM method with other SR method for upsampling factor $\times 4$ and $\times 8$ on *noisy* depth image *Art* and *Baby* respectively. Bottom row of Figure 6.12 show results obtained from SRGMM method using direct and hierarchical approaches for 100 Gaussian mixture. The zoomed portion of the image is shown



(a) GT (b) Bic (c) ATGV (Ferstl et al., 2013)



(d) GIF (He et al., 2010) (e) SRGMM-Dir 100Mix (f) SRGMM-Iter 100Mix

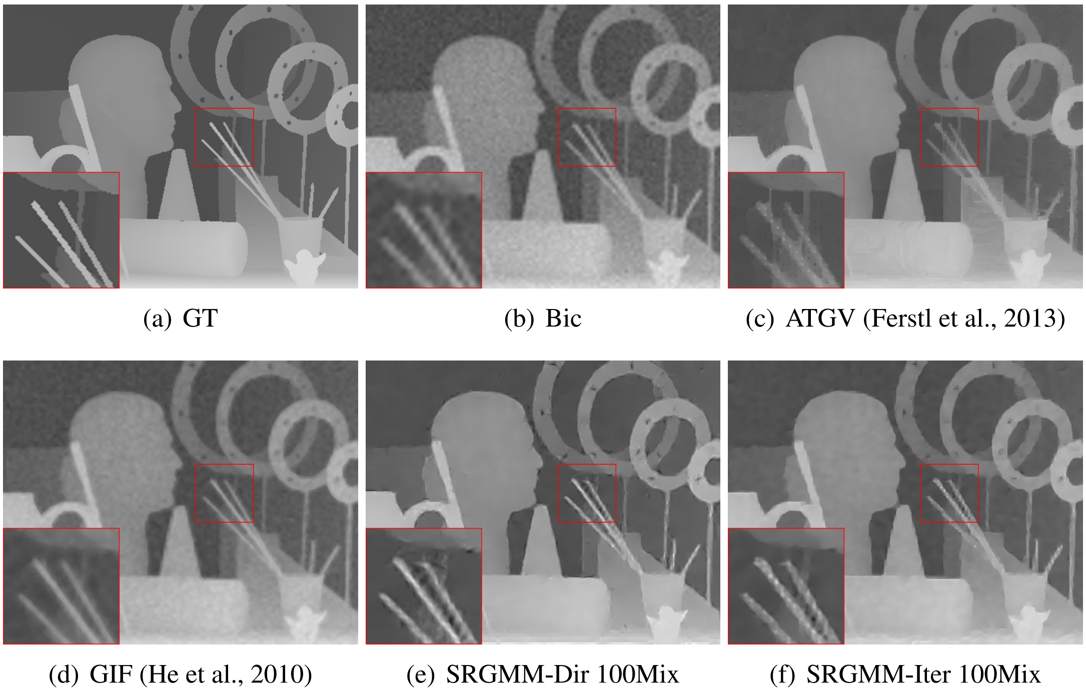
Figure 6.11: Qualitative results comparison for SR by factor $\times 8\_n0$ (Image: *Baby*)

Table 6.8: PSNR/SSIM result comparison of SR for factor $\times 2$, $\times 4$ and $\times 8$ on *noiseless* images (Red text indicate highest value and blue text indicate second highest value)

Images	Bil	Bic	ATGV	GIF	RI	SRGMM-Dir 50Mix	SRGMM-Hier 50Mix	SRGMM-Dir 100Mix	SRGMM-Hier 100Mix
Aloe	33.36/0.95	33.67/0.95	34.41/0.96	33.83/0.96	35.42/0.97	37.58/0.98	-	37.96/0.98	-
Art	31.02/0.92	31.42/0.93	32.01/0.94	31.70/0.93	32.76/0.95	35.25/0.97	-	35.52/0.97	-
Baby	37.95/0.98	38.27/0.98	39.16/0.98	38.44/0.98	40.15/0.99	43.54/0.99	-	43.70/0.99	-
Cones	36.25/0.97	36.58/0.97	37.50/0.97	36.75/0.97	38.09/0.98	39.53/0.98	-	40.00/0.98	-
Plastic	39.28/0.99	39.57/0.99	41.64/0.99	40.14/0.99	41.72/0.99	44.92/0.99	-	45.76/0.99	-
Reindeer	34.17/0.96	34.51/0.97	35.10/0.97	34.87/0.97	35.99/0.98	38.40/0.98	-	38.64/0.98	-
Sawtooth	37.50/0.98	37.86/0.98	38.76/0.98	38.71/0.98	39.51/0.99	43.93/0.99	-	44.13/1.00	-
Venus	42.35/0.99	42.69/0.99	43.87/0.99	43.25/0.99	44.48/0.99	48.05/1.00	-	48.90/1.00	-
Avg. x2_n0	36.48/0.96	36.82/0.97	37.80/0.97	37.21/0.97	38.51/0.98	41.40/0.98	-	41.82/0.98	-
Aloe	30.10/0.92	30.11/0.92	31.90/0.94	30.47/0.93	34.82/0.97	34.82/0.96	33.42/0.96	34.72/0.96	35.66/0.97
Art	28.26/0.88	28.44/0.88	29.80/0.91	28.87/0.89	31.61/0.94	32.19/0.93	27.53/0.94	31.83/0.93	31.97/0.94
Baby	34.78/0.97	34.87/0.97	36.89/0.98	35.19/0.97	39.25/0.99	39.41/0.98	37.67/0.98	39.36/0.98	39.71/0.99
Cones	33.08/0.95	33.12/0.95	35.31/0.96	33.56/0.95	36.83/0.97	36.60/0.97	36.51/0.97	36.69/0.97	35.88/0.97
Plastic	36.27/0.98	36.33/0.98	41.55/0.99	36.76/0.98	38.47/0.99	41.45/0.99	37.32/0.98	38.65/0.98	41.87/0.99
Reindeer	31.16/0.94	31.25/0.94	34.01/0.97	31.73/0.95	35.12/0.97	35.04/0.96	34.12/0.97	35.09/0.96	34.65/0.97
Sawtooth	34.84/0.97	35.04/0.97	37.61/0.98	35.80/0.97	38.76/0.99	39.66/0.99	36.76/0.99	40.24/0.99	37.19/0.99
Venus	39.63/0.98	39.81/0.98	42.62/0.99	40.43/0.99	44.54/0.99	44.39/0.99	43.13/0.99	44.23/0.99	39.27/0.99
Avg. x4_n0	33.51/0.94	33.62/0.94	36.21/0.96	34.10/0.95	37.42/0.97	37.94/0.97	35.80/0.97	37.60/0.97	37.02/0.97
Aloe	26.46/0.88	26.17/0.87	26.12/0.90	26.36/0.88	30.27/0.92	29.32/0.89	30.11/0.92	29.54/0.90	30.45/0.93
Art	24.84/0.81	24.69/0.81	25.95/0.86	24.91/0.82	28.05/0.88	26.26/0.83	26.48/0.87	26.05/0.83	26.02/0.87
Baby	30.64/0.94	30.45/0.94	32.52/0.96	30.67/0.94	36.35/0.97	34.34/0.96	33.43/0.97	33.14/0.95	34.99/0.96
Cones	29.79/0.92	29.59/0.92	30.72/0.93	29.79/0.93	32.63/0.95	30.49/0.92	32.11/0.95	30.45/0.92	31.02/0.94
Plastic	31.11/0.96	30.93/0.96	33.46/0.97	31.08/0.96	36.39/0.98	34.23/0.97	29.35/0.97	33.31/0.97	36.36/0.97
Reindeer	27.91/0.92	27.73/0.91	29.64/0.94	28.02/0.92	31.39/0.95	28.60/0.92	29.94/0.93	28.94/0.92	29.43/0.94
Sawtooth	31.25/0.95	31.20/0.95	32.21/0.96	31.58/0.95	34.23/0.97	33.11/0.96	33.45/0.97	32.85/0.96	34.49/0.97
Venus	36.06/0.97	35.98/0.97	38.61/0.97	36.36/0.97	41.11/0.99	36.89/0.98	39.83/0.98	36.96/0.98	35.54/0.98
Avg. x8_n0	29.75/0.91	29.59/0.91	31.15/0.93	29.84/0.92	33.80/0.95	31.65/0.92	31.83/0.94	31.40/0.92	32.28/0.94

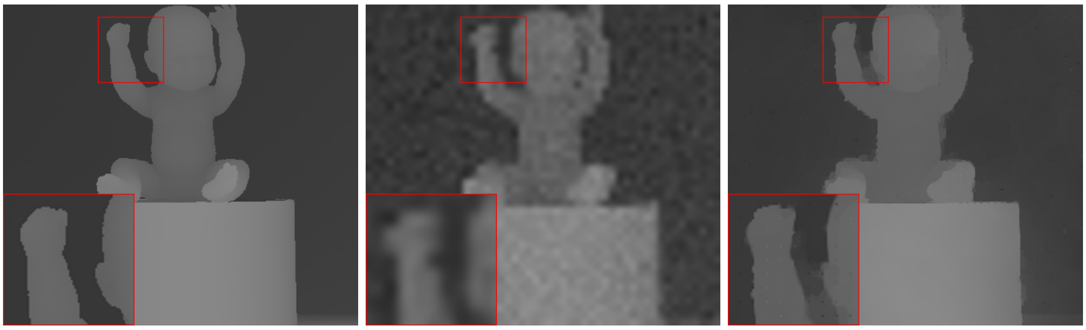
inset to the image, and we can see that the sticks in *Art* image produced by SRGMM method (both direct and hierarchical approach) has better depth discontinuities and the smoother regions are much smoother with lesser noise. Figure 6.13 show results of SR by factor $\times 8$, where hierarchical approach performs better than the direct approach in terms of edge preservation and noise smoothing.

Table 6.9 shows the PSNR and SSIM values obtained from the SR methods on *noisy* depth images. Although the ATGV method performs well for the noisy images, but SRGMM method, on an average, with 50 Gaussian mixture performs marginally better than ATGV method. In particular, on an average, we preform 2.80 dB and 3.25 dB better than bilinear and bicubic interpolation methods respectively, and perform about 0.64 dB, 2.90 dB and 0.53 dB better than ATGV, GIF and RI methods respectively in $\times 8\_n5$ case. Overall, the hierarchical approach performs better then the contemporary methods for higher upsampling factors like $\times 4$ and $\times 8$ (best average value in red colour). Although direct approach does better, the hierarchical approach does superior as it learns more finer mapping between the HR-LR patches in the iterative process of upsampling by factor 2 to reach the higher upsampling factors like 4 and 8.

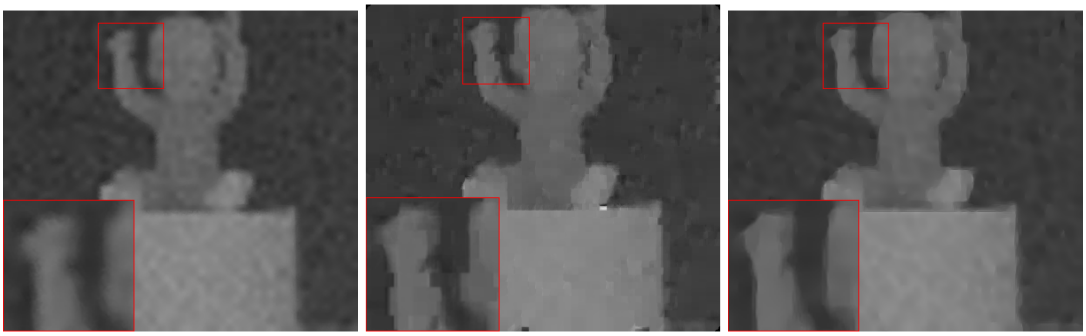

(a) GT (b) Bic (c) ATGV (Ferstl et al., 2013)

(d) GIF (He et al., 2010) (e) SRGMM-Dir 100Mix (f) SRGMM-Iter 100Mix

Figure 6.12: Qualitative results comparison for SR by factor $\times 4\_n5$ (Image: *Art*)



(a) GT (b) Bic (c) ATGV (Ferstl et al., 2013)



(d) GIF (He et al., 2010) (e) SRGMM-Dir 100Mix (f) SRGMM-Iter 100Mix

Figure 6.13: Qualitative results comparison for SR by factor $\times 8\_n5$ (Image: *Baby*)

Table 6.9: PSNR/SSIM result comparison of SR for factor $\times 2$, $\times 4$ and $\times 8$ on *noisy* images (Red text indicate highest value and blue text indicate second highest value)

Images	Bil	Bic	ATGV	GIF	RI	SRGMM-Dir 50Mix	SRGMM-Hier 50Mix	SRGMM-Dir 100Mix	SRGMM-Hier 100Mix
Aloe	32.15/0.87	31.70/0.81	32.56/0.85	33.12/0.92	32.51/0.82	35.48/0.96	-	35.60/0.96	-
Art	30.27/0.84	30.12/0.79	30.96/0.85	31.24/0.90	30.86/0.80	33.66/0.95	-	33.70/0.95	-
Baby	35.05/0.88	33.94/0.82	35.54/0.88	36.68/0.94	34.38/0.82	37.09/0.94	-	36.69/0.93	-
Cones	34.16/0.88	33.33/0.82	34.56/0.87	35.54/0.94	33.83/0.83	37.73/0.97	-	37.94/0.97	-
Plastic	35.75/0.89	34.44/0.82	37.01/0.89	38.00/0.95	34.85/0.82	41.61/0.98	-	41.77/0.98	-
Reindeer	32.76/0.87	32.21/0.81	33.77/0.88	34.10/0.94	32.86/0.82	36.56/0.97	-	36.59/0.97	-
Sawtooth	34.81/0.88	33.78/0.81	35.15/0.87	36.96/0.94	34.17/0.81	37.80/0.95	-	37.35/0.94	-
Venus	36.77/0.88	35.11/0.82	36.83/0.87	39.48/0.95	35.27/0.82	39.91/0.96	-	39.22/0.95	-
Avg. x2_n5	33.96/0.87	33.07/0.81	34.54/0.87	35.64/0.93	33.59/0.81	37.48/0.96	-	37.35/0.95	-
Aloe	29.39/0.87	29.08/0.84	31.24/0.91	29.76/0.88	32.25/0.88	33.27/0.93	33.12/0.94	33.06/0.93	33.45/0.94
Art	27.79/0.83	27.72/0.80	29.41/0.89	28.36/0.85	30.19/0.85	31.27/0.91	30.60/0.91	30.99/0.91	30.67/0.91
Baby	33.08/0.90	32.41/0.87	35.77/0.95	33.54/0.92	34.36/0.89	34.88/0.91	35.95/0.95	34.55/0.91	35.67/0.95
Cones	31.90/0.89	31.39/0.86	34.31/0.93	32.44/0.91	33.39/0.88	35.07/0.95	34.91/0.95	35.26/0.95	34.88/0.95
Plastic	34.01/0.91	33.18/0.87	39.20/0.97	34.63/0.93	34.55/0.89	39.10/0.98	38.17/0.97	37.95/0.97	37.85/0.97
Reindeer	30.31/0.88	30.00/0.85	33.22/0.94	30.91/0.91	32.44/0.88	33.81/0.95	33.39/0.95	33.83/0.95	33.67/0.95
Sawtooth	33.03/0.90	32.43/0.86	35.98/0.95	33.92/0.92	33.99/0.88	35.59/0.93	36.00/0.96	35.50/0.93	36.01/0.96
Venus	35.57/0.91	34.43/0.87	39.14/0.95	36.42/0.94	35.39/0.89	37.59/0.94	38.50/0.96	37.14/0.94	38.00/0.96
Avg. x4_n5	31.88/0.88	31.33/0.85	34.78/0.93	32.49/0.90	33.32/0.88	35.07/0.93	35.08/0.94	34.78/0.93	35.02/0.94
Aloe	26.16/0.86	25.75/0.84	25.87/0.89	25.98/0.85	29.40/0.89	29.12/0.88	29.61/0.90	28.98/0.88	29.83/0.90
Art	24.62/0.79	24.38/0.78	25.87/0.86	24.62/0.79	27.25/0.84	25.86/0.81	26.85/0.84	25.69/0.82	26.75/0.84
Baby	29.85/0.92	29.37/0.90	32.31/0.96	29.69/0.91	32.62/0.93	32.06/0.92	33.02/0.94	31.43/0.92	32.92/0.94
Cones	29.24/0.90	28.82/0.89	30.45/0.93	29.12/0.90	31.12/0.91	30.15/0.91	31.42/0.93	30.16/0.91	31.44/0.93
Plastic	30.28/0.94	29.77/0.92	33.81/0.97	30.07/0.93	32.97/0.94	33.10/0.95	34.89/0.96	32.27/0.95	31.43/0.95
Reindeer	27.51/0.89	27.17/0.88	29.56/0.94	27.52/0.89	30.00/0.91	28.31/0.90	30.10/0.92	28.43/0.90	30.01/0.92
Sawtooth	30.38/0.92	29.97/0.91	32.56/0.96	30.46/0.92	31.87/0.93	30.81/0.92	32.51/0.94	31.37/0.93	31.38/0.94
Venus	33.74/0.94	32.95/0.93	38.67/0.98	33.54/0.94	34.71/0.94	33.97/0.94	35.78/0.96	33.76/0.94	34.19/0.96
Avg. x8_n5	28.97/0.89	28.52/0.88	31.13/0.93	28.87/0.89	31.24/0.91	30.42/0.90	31.77/0.92	30.26/0.90	30.99/0.92

6.4.5 SR Results on Real Depth Images

We now demonstrate the SRGMM method on real-time depth images taken from modern ToF depth cameras. The source of these real-time depth images is from Ferstl et al. (2013). These images are *books*, *shark* and *devil*, each of size 120×160. These images are challenging as they represent the real depth images taken from the modern depth cameras in a real situation which has obvious problems like low spatial resolution and noise corrupted. Using the learned models, we upsample these ToF depth images by a factor of $\times 2$, $\times 4$ and $\times 8$, and we show the SR results of SRGMM method in comparison with bicubic interpolation results.

Figure 6.14 shows the SR results produced by SRGMM method for SR factor 2, 4 and 8 on *books*, *shark* and *devil* ToF depth images. The first row shows the ToF LR depth images, and the second, third, and fourth row shows the SR results from SRGMM method for upsampling factor $\times 2$, $\times 4$ and $\times 8$ respectively. The results shown here are produced from a GMM model trained over 300 Gaussian mixtures. We can observe in second row (for SR by factor 2) that the outputs produced are much smoother and have preserved edges to a larger extent. The level of noise has definitely reduced by a large margin as one can see in the cropped and zoomed portion of these images in Figure 6.15. The third and the fourth row (for SR factor 4 and 8 respectively) is special, as it demonstrate the SR results for higher upsampling factor. Inspite of having higher target resolution for SR factors 4 and 8, the output images are smoother with well preserved edges and the overall image structure. As compared to bicubic interpolation results, which blurs the edges details and heavily degraded with noise, SRGMM method reduce noise to a larger extent and the edges discontinuities are also maintained in all the upsampling factor of $\times 2$, $\times 4$ and $\times 8$. This is because, SRGMM method works on patch-based methods to learn the HR-LR relationship, whereas the interpolation methods have implicit smoothing constraint which blurs the prominent edges without giving it much importance.

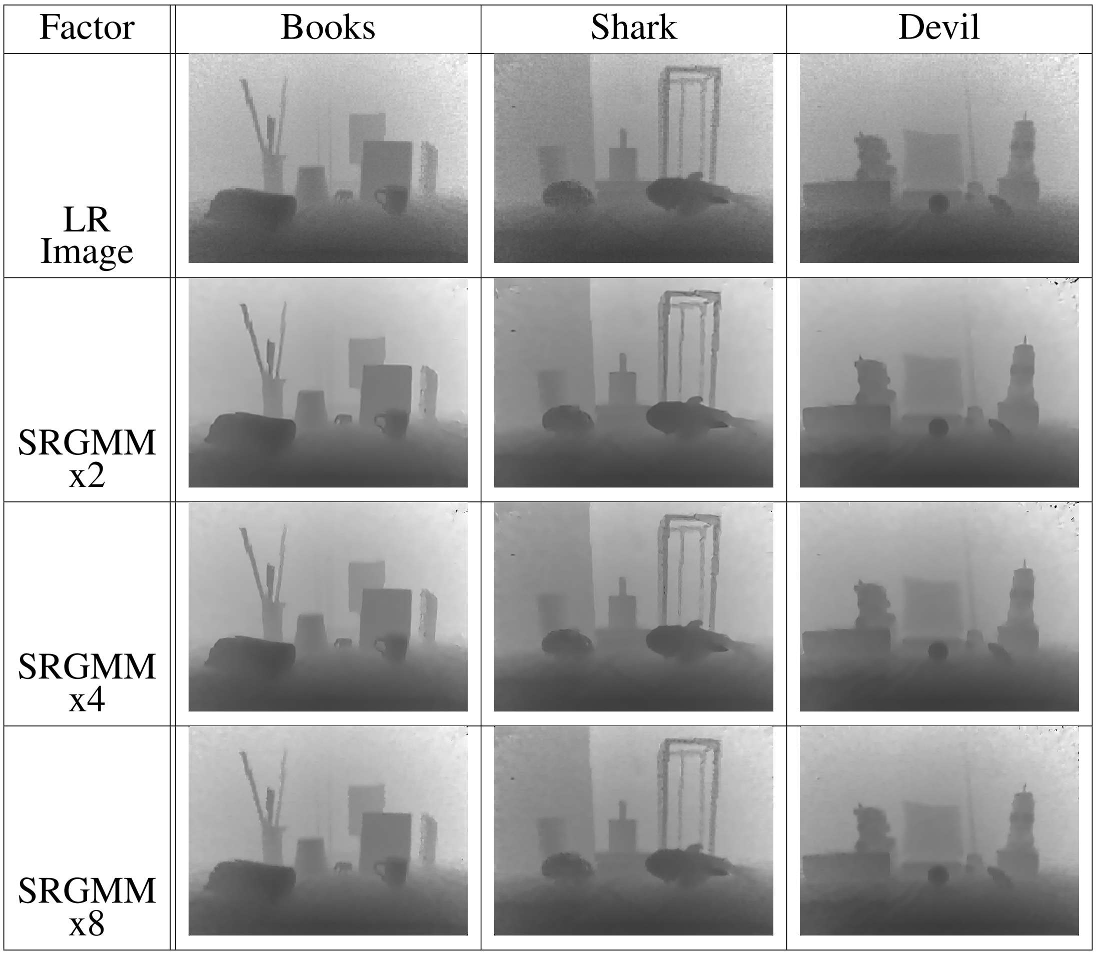


Figure 6.14: SR results of SRGMM on real tof depth images.

6.4.6 Time Complexity

From the standard GMM training point of view, the training procedure consume maximum time of the total SRGMM time. The important part of any method is its test time, which tells us how effective a method is in producing the desired output. We have tested SRGMM and all other comparative methods on 64-bit windows OS desktop with CPU configuration Intel(R) Core(TM) i7-3770 CPU @ 3.40GHz with 8GB RAM using 64-bit Matlab software R2015a (8.5.0.197613).

Table 6.10 shows the average time complexity (in sec.) of various SR methods computer on the chosen test set. We show the timing analysis only for SR factor 2 on *one-third* size target resolution images. We present the timings of SRGMM model trained over 100 and 500 Gaussian mixtures only. The representation 10L_[8,4] means

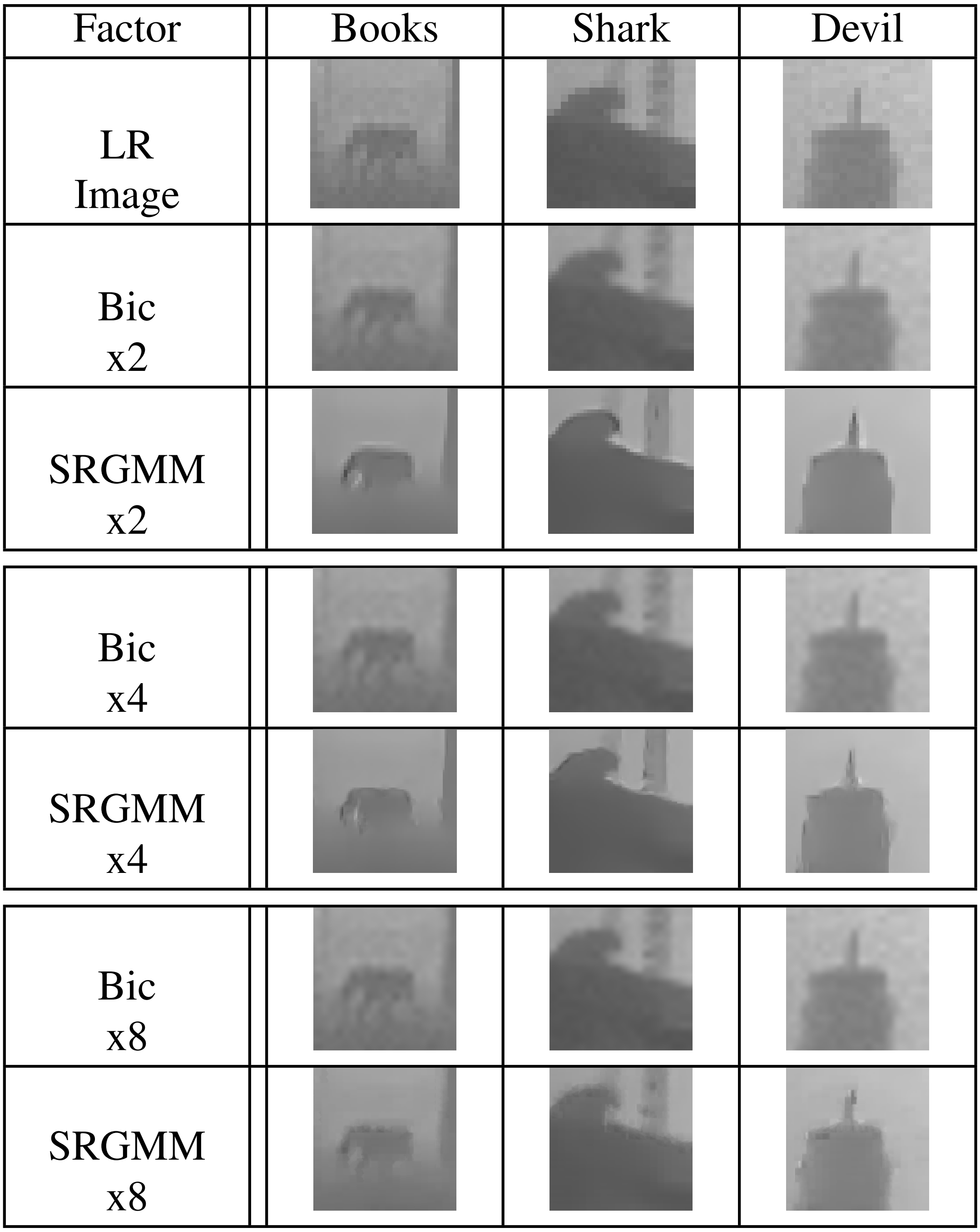


Figure 6.15: Cropped region of SR results of SRGMM on real tof depth images.

that the proposed SRGMM model trained with 10 Lakhs of vectors with HR-LR patch sizes as 8×8 and 4×4 respectively.

Bicubic interpolation method takes the least time to compute the SR output as it has no learning involved in its procedure. From 10L_[8,4] to 10L_[6,3] to 10L_[4,2] the execution time is gradually reducing because of the reduced HR-LR patch sizes, and from 5L_[8,4] to 2.5L_[8,4] to 1L_[8,4] the execution time is almost same as the HR-LR patch sizes is kept same. Even though the total number of training vectors are reduced from 10L to 1L, the execution time is almost similar with HR-LR patch sizes of [8,4]. However, the performance with 1L_[8,4] is nealy equal to 10L_[8,4] as we do not require many training vectors for depth images as depth images do not have much variations and are smoother in most of the regions.

Table 6.10: Time complexity for SR Factor ×2 (in sec.)

	SR by factor x2									
	Bic	GIF	ATGV	RI	Proposed SRGMM (100GMM / 500GMM)					
					10L_[8,4]	10L_[6,3]	10L_[4,2]	5L_[8,4]	2.5L_[8,4]	1L_[8,4]
Avg. time	0.0015	2.60	59.81	0.075	1.15 / 4.15	0.89 / 2.65	0.75 / 1.55	1.25 / 4.22	1.24 / 4.30	1.27 / 4.35

6.5 SUMMARY

We have used a popular method of parametric probability density function estimation called Gaussiam mixture model (GMM) for the purpose of single depth image super-resolution. The use of GMM is already popular in the areas of speech recognition and many image processing tasks like image segmentation, image denoising, image super-resolution, and more. The GMM model is trained from synthetic images having sharp edges and varying depth values. For most of the test images, Gaussian mixtures between 200-250 gave good results, because it is enough to separate the similar looking vectors in the vector space, and it can capture the HR-LR relationship quite well.

To show the effectiveness of the target resolution, we have have performed the experiments with *one-third* and *full* size images, and found improvements in the SRGMM results for *full* size image. We have also demonstrated SRGMM method trained with different HR-LR patches sizes and seen that [8,4] patch combination does better job of preserving the edge discontinuities and also reduces noise to a much lower level. However, [4,2] patch combination does poor than other combination because there will be many similar LR patches in a selected Gaussian component for an input LR patch, and the one with lowest MMSE is selected to retrieve the corresponding HR patch, and a small pixel different in the selected LR patch might reflect major differences in the HR patch.

The training image based SR methods will give better results. However the guidance based method can also improve upon the results if there is better initial estimate for the SR output. For iterative SR method, the SR initial estimate is very critical in deciding the convergence of the solution. Hence, the next chapter is motivated by the concept of having a better initial estimate which is as computationally low as bicubic interpolation for faster convergence.

CHAPTER 7

BETTER INITIAL ESTIMATE FOR ITERATIVE SUPER-RESOLUTION METHODS

[1] As discussed earlier, SR is an ill-posed inverse problem. From an input LR image with few known pixel values, the SR method super-resolves it to a higher spatial resolution image, which makes the system under-determined which will have either no solution or have infinite solutions. A better initial HR estimate of the output is kind of good regularizer to find the optimal solution in an infinite solution space. A SR pipeline for a better initial estimate is proposed, especially for higher upsampling factor, which will help in quick convergence and lead to an accurate output. The proposed SR pipeline is a cascade approach of two methods, where the first method in the SR pipeline produces a better initial estimate for the cascaded method following it in a pipeline to improve upon. The Residual Interpolation (RI) method which is as fast as standard bicubic interpolation method, and produces better output image with sharp boundaries, can be better suited for producing an initial HR estimate for a given LR input image. The RI output is then fed to an iterative method to improve the image details. The Anisotropic Total Generalized Variation (ATGV) method has been used as a second module in the cascaded SR pipeline. Both RI and ATGV method require corresponding registered HR guidance image for its operation. The improvements are shown qualitative and quantitative on depth images from Middlebury dataset for different upsampling factors (i.e. $\times 2$, $\times 4$ and $\times 8$), and with different levels of noise.

---

[1] Chandra Shaker Balure, M. Ramesh Kini . ”Guidance Based Improved Depth Upsampling With Better Initial Estimate.” *Inderscience, International Journal of Computational Vision and Robotics.* **[Under Review]**

7.1 INTRODUCTION

SR problem is an ill-posed inverse problem, hence they are regularized to obtain an optimal solution from the infinite solution space. Since depth images are mostly smooth or linearly smooth at object surfaces and the sharp discontinuity at object boundaries, the SR methods can target for larger upsampling factor, thereby retaining the edge discontinuities and depth precision to a large extent. In literature, there are variety of methods which have tried to address single depth image super-resolution problem. Most of the SR methods try to first estimate an initial HR output, where it is treated as an initial estimated solution. The initial estimation process varies across different SR methods, where some methods start with a sparse LR input mapped on a HR grid of the desired resolution (Ferstl et al., 2013) as an initial estimate of the solution, or some methods start with bicubic interpolation of the LR input (Yang et al., 2013) as an initial solution. For higher upsampling factors, the initial estimate from LR image to HR grid mapping will lead to numerous unknown pixels which needs to be determined. As the ratio of known to unknown pixels is very high, the SR method suffer from over-smoothing artifacts in the output image. Similarly, the initial estimate computed using bicubically interpolation method also suffer from over-smoothing artifacts because the interpolation method will have to estimate multiple unknown (depends on upsampling factor) between the known points, and the bilinear or bicubic approach produces smoothened image.

In this chapter, a suitable approach for better initial estimate is presented which is as fast as classical interpolation, and as good as other SR methods. The task becomes more challenging with added noise, which is what has been considered in this work. The proposed method combines two distinct methods, i.e. residual interpolation (RI) method (Konno et al., 2015) and anisotropic total generalized variation (ATGV) method (Ferstl et al., 2013). The proposed method falls under the category of guidance based depth super-resolution method where the proposed method combines RI method (Konno et al., 2015) and ATGV method (Ferstl et al., 2013), and both these methods require HR intensity image as guidance image for their functioning. The RI method is utilized to generate an initial estimate (which is fast and easy), and then employed the ATGV method for final depth restoration (which is efficient). As the upsampling factor goes higher from 2 to 4 to 8 and more, the proposed combination of RI and ATGV does a good job of producing a satisfying upsampled depth image.

The proposed method has been tested on standard dataset from Middlebury (Scharstein and Szeliski, 2003) for different upsampling factors (i.e. $\times 2$, $\times 4$ and $\times 8$). The experimental results for noisy images are shown qualitatively and quantitatively, and these results are compared against classical interpolation methods (bilinear and bicubic), and also against RI method (Konno et al., 2015) and ATGV method (Ferstl et al., 2013) when treated individually.

7.2 BACKGROUND DETAILS

In the following sections, two modules which are integral part of the SR pipeline in the proposed work are discussed, i.e. the RI module (Konno et al., 2015) and ATGV module (Ferstl et al., 2013).

7.2.1 Brief Description of Residual Interpolation Method

RI method (Konno et al., 2015) is the initial module in the SR pipeline. It takes the LR depth input d and the HR guidance image I. The complete flow of RI method is shown in the block diagram in Figure 7.1, where the LR depth image is represented by d and the HR colour guidance image represented by I. The complete process of RI is mainly processed in residual domain, where *residual* means the difference between the tentative depth output and the LR input. The tentative depth output is generated using the popularly known guided image filtering (GIF) (He et al., 2010) approach, which considers that the dominant edges in the input depth image coincides with the edges in the colour guidance image by considering the local linear relationship between d and I. The local linear combination between the input guidance image and the tentative output

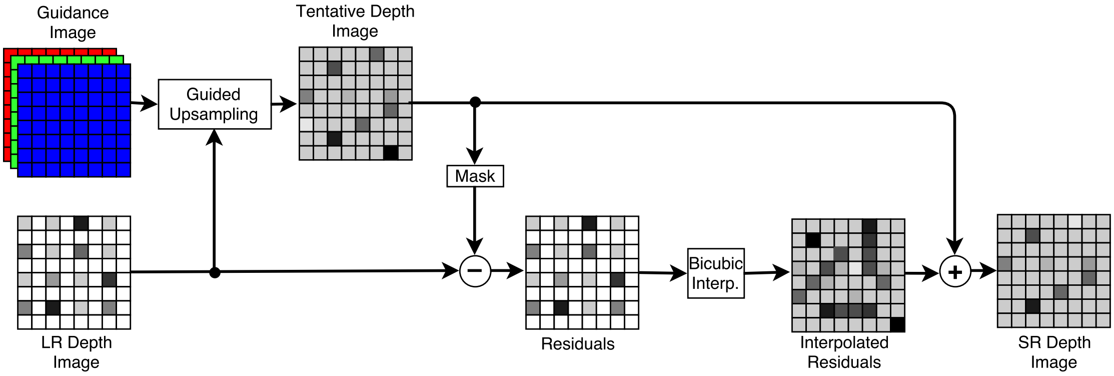


Figure 7.1: Block diagram of residual interpolation method Konno et al. (2015)

is given in Eq. 7.1,

$$t_i = a_k I_i + b_k, \quad \forall i \in w_k, \tag{7.1}$$

where, I and t represent the HR colour guidance image and the HR tentative depth output, and i represent all the pixel location in those images, and w_k denotes local window centered at pixel location k, and a_k and b_k are the local linear coefficients. These linear coefficients for each pixel location at calculation is calculated by minimizing the cost function $E(\cdot)$ which is given in Eq. 7.2,

$$E(a_k, b_k) = \sum (a_k I_i^M + b_k - d_i^2) + \eta a_k^2 \tag{7.2}$$

where, I_i^M is the pixel value of the masked HR intensity image, and d_i is the corresponding LR depth value, and η is the regularization parameter. The linear coefficients for a pixel location are obtained by weighted averaging given in Eq. 7.3, instead of just averaging,

$$\hat{a}_k = \frac{\sum_{i \in w_k} W_i a_i}{\sum_{i \in w_k} W_i}, \quad \hat{b}_k = \frac{\sum_{i \in w_k} W_i b_i}{\sum_{i \in w_k} W_i} \tag{7.3}$$

where, the weight W is determined by the cost of GIF as in Eq. 7.4,

$$W_i = \frac{1}{\max\left(\frac{1}{|\omega_i|} \sum (a_i I_i^M + b_i - d_j)^2, \delta\right)} \tag{7.4}$$

where, δ is the threshold parameter to avoid the *divide-by-zero* situation. The tentative estimate is finally calculated as given in Eq. 7.5,

$$t_i = \hat{a}_k I_i + \hat{b}_k, \quad \forall i \in w_k, \tag{7.5}$$

The tentative output is then masked in accordance to the sparse LR depth input to obtain the residuals. The residual is the result of the image difference between the the masked tentative output and the sparse LR input. The tentative output will be sharper than the LR input, and hence their difference will give us the high-frequency information. The residual image is then bicubic interpolated to estimate the missing pixels from the residual image grid. The interpolated image is then added back to the tentative estimated depth image of earlier step to get the final output image which is clear and plausible with sharp edge discontinuities.

The RI output is more accurate in terms of the edge sharpness and depth precision, and hence it is used on the proposed SR pipeline as a better initial estimate. There are few benefits of considering RI output as an input to the next cascaded ATGV module. Firstly, with good initial solution, the convergence will be faster. Secondly, the output will be more accurate as opposed to the approaches which used no such initial estimate, instead they start from the sparse LR depth input itself as in the case of ATGV Ferstl et al. (2013) method.

The RI depth output along with the same HR colour guidance image is passed as input to ATGV module to obtain the super-resolved output depth image. Even for higher magnification factor, the initial solution from RI method does a better job of preserving the edge information and converge to the solution much faster.

7.2.2 Brief Description of Anisotropic Total Generalized Variation Method

ATGV method (Ferstl et al., 2013) was initially proposed for depth image super-resolution problem by keeping in mind the shortcoming of modern depth cameras. As discussed earlier, there are variety of modern depth cameras which can capture depth images based on the principle of time-of-flight (ToF). The ATGV method tries to solve one of the problems of such modern depth cameras, i.e. low spatial resolution problem. ATGV method uses variational optimization framework to super-resolve the LR image by adding information from the HR guidance image. The complete work flow is shown in the block diagram in Figure 7.2. Ferstl et al. (2013) proposed a convex optimization problem which has two terms involved in it, one is the data term and second one is the regularization term. The data term enforces the output to look similar to the input measurements, and the regularization term enforces piecewise solution by preserving the edges and reducing the noise. The regularization term, they use higher order total generalized variation (TGV) regularization which is weighted according to the texture in the intensity image by an anisotropic diffusion tensor.

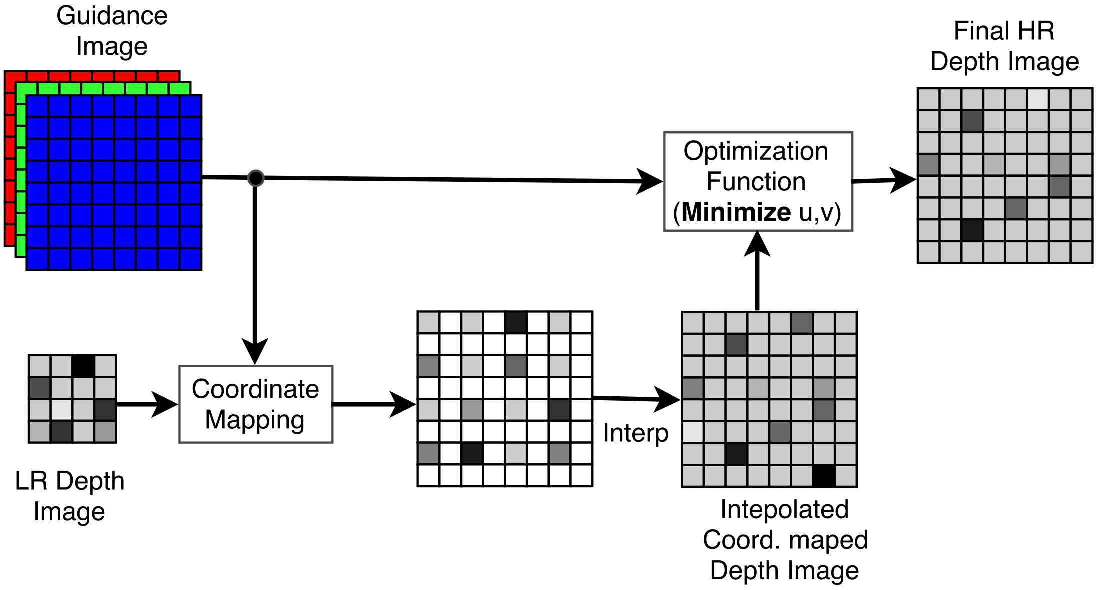

Guidance Image
Final HR Depth Image
Optimization Function (**Minimize** u,v)
Coordinate Mapping
Interp
LR Depth Image
Intepolated Coord. maped Depth Image

Figure 7.2: Block diagram of anisotropic total generalized variation method Ferstl et al. (2013)

With the formation of core convex energy functional, the whole upsampling process is divided into three steps, which are:

1. First task is to register the LR depth image and HR guidance image into one common coordinate system.
2. Then formulating the convex energy function with higher order regularization function.
3. Then solving the optimization function with first-order primal-dual optimization scheme.

For coordinate mapping, one image plane has to be considered as a reference plane on which the other image is projected back. Here, the HR guidance image plane is considered as a reference plane with known intrinsic and extrinsic camera parameters. The LR depth image d at each pixel location $x_{i,j} = [i, j, 1]^T$ is projected onto the HR image plane to a new 3D pixel location $\hat{x}_{i,j}$, which is represented as in Eq. 7.6,

$$\hat{d}_{i,j} = C_L + d_{i,j} \frac{P_L^{\dagger} x_{i,j}}{\parallel P_L^{\dagger} x_{i,j} \parallel} \tag{7.6}$$

where C_L is the depth camera center and $P_L^{\dagger}$ is the pseudoinverse of depth camera projection matrix. This projected image gives the sparse HR depth image as the mapping from depth image space to guidance image space is on-to-one to avoid the problem of averaging, whereas the unknown pixels are interpolated.

From the sparse HR depth image and with additional cue from HR guidance image, the dense HR depth image is give by Eq. 7.7,

$$\hat{D} = \arg\min_{u} \{G(u, \hat{d}) + \alpha F(u)\} \tag{7.7}$$

where $G(u, \hat{d})$ is the data term that measures quality of u to the input $\hat{d}$, and $F(u)$ is the regularization term with prior knowledge of smoothness of the final solution, and G and F are the convex lower semi-continuous functions, and α variable is to balance between the data term and the regularizer. The data term is represented by Eq. 7.8 as,

$$G(u, D_S) = \int_{\Omega_H} w|u - \hat{d}|^2 dx \tag{7.8}$$

where, w is a weighter operator between $[0, 1] \in \mathbb{R}_{\Omega_H}$ which is zero at unmapped image points.

The whole burden is on the regularization term to produce a sharp depth output. Earlier, regularization terms were of first-order smoothness, for example total variation semi norm with L1 norm gives $\| \nabla u \|_1$, but this regularizer could not be used for depth images resulting in piecewise fronto parallel depth reconstruction. Hence, a more generalized regularization model called Total Generalized Variation (TGV) is used, which is composed of polynomials of arbitrary order which results in piecewise polynomial depth reconstruction. An order of k favors solutions composed of polynomials of order $k-1$, so for depth images second-order TGV suffice, which is given by Eq. 7.9,

$$\text{TGV}_\alpha^2 = \min_v \{\alpha_1 \int_\Omega |\nabla u - v| dx + \alpha_0 \int_\Omega |\nabla v| dx\} \tag{7.9}$$

where α_0 and α_1 are scalar weights. To produce accurate HR depth output at the edge discontinuities, an anisotropic diffusion tensor $T^{\frac{1}{2}}$ is computed using the HR guidance image, which is calculated as shown by Eq. 7.10,

$$T^{\frac{1}{2}} = \exp(-\beta |\nabla I_H|^\gamma) n n^T + n^\perp n^{\perp T} \tag{7.10}$$

where n is the direction of the gradient, and $n^\perp$ is the normal vector to the gradient, and β and γ adjust the direction and sharpness of the tensor.

The final energy is defined as a combination of data term (Eq. 7.8) and TGV term (Eq. 7.9) with anisotropic diffusion (Eq. 7.10) is represented in Eq. 7.11 as,

$$\begin{aligned} \min_{u,v} \{\alpha_1 \int_{\Omega_H} & |T^{\frac{1}{2}} (\nabla u - v)| dx + \\ \alpha_0 \int_{\Omega_H} & |\nabla v| dx + \\ \int_{\Omega_H} & w |u - \hat{d}|^2 dx|\} \end{aligned} \tag{7.11}$$

To find the solution to this convex optimization problem, they use primal-dual energy minimization scheme which runs iteratively for all pixels individually.

7.3 PROPOSED METHOD

As like any other super-resolution problem, here also the the problem statement is quite similar, except that an extra input of guidance HR colour image is present. Under the guidance of this HR colour image I, the SR method is required to take an LR depth image d to produce an HR depth image $\hat{D}$ which must be as close to the ground truth image D as possible. The spatial resolution of the guidance image is equal to the required output resolution for the input image.

Like any other guidance image based SR methods, here also it is assumed that the input LR depth image and the HR colour image are co-aligned at each pixel. This assumption is valid, as it is seen in the literature that capturing the intensity image is a low-cost operation and easy, and it can be captured along with the depth images mounted on a same rig.

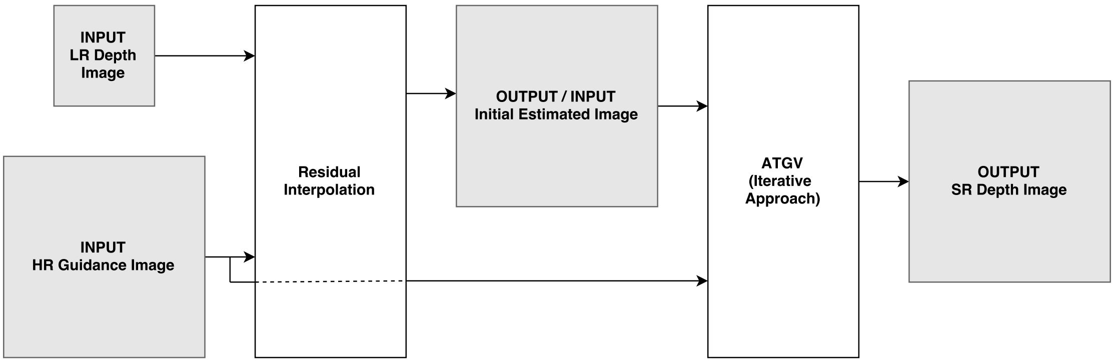


Figure 7.3: Block diagram of proposed method combining RI and ATGV in cascade form.

The overall proposed method is shown as block diagram in Figure 7.3. This figure shows the complete work flow of the proposed method, which is basically a cascade of two approaches combined in a single framework to get a sharp and accurate HR depth output. To this whole SR pipeline, two inputs are fed, where one is the LR depth image and other one is the HR guidance image. The proposed SR pipeline is composed of two methods cascaded in a single framework to produce an HR output. The residual interpolation (RI) method (Konno et al., 2015) is inspired by GIF approach, where it assumes the local linear mapping between the guidance image and the output image. RI method operates in the residual domain, where the *residual* is the different between the tentative estimated HR depth map and the LR depth map. This *residual* is then interpolated using standard interpolation method, and then added it to the tentative estimated HR depth map to recover the final HR depth map. As this process is easy and fast, it is considered as a first module in the proposed SR pipeline. The RI output is considered as a better initial estimate which is better in terms of preserving the edge discontinuities in the image.

The output image from RI module is then fed as an input to the anisotropic total generalized variation (ATGV) module (Ferstl et al., 2013) which acts a a cascade module in the proposed SR pipeline. ATGV module also makes use of the same HR guidance image as used in the first stage. ATGV use anisotropic diffusion tensor, calculated from HR guidance image, which is used to guide the upsampling process.

7.4 EXPERIMENTAL RESULTS AND DISCUSSIONS

The proposed HR guidance image based depth image super-resolution method has been evaluated on depth image from standard dataset of Middlebury (Scharstein and Szeliski, 2003). This dataset is chosen because it has both the depth image and its corresponding registered colour images. The proposed method is demonstrated on noisy inputs with different levels of added noise with noise standard deviation ranging from $\sigma = 1, 2, 3, 4$ and 5). The results are shown for three different upsampling factor i.e. 2, 4 and 8.

The LR image used for testing purpose were generated using the LR image model, which was discussed in Chapter 1. The LR image model is used only generating the observed LR image, and it has not been used in the reconstructing of the output. The SR results of proposed method (ATGVMod) is compared with the RI method and ATGV method as state-of-the-art guidance based depth SR methods, and it is also compared with classical bicubic interpolation method. To show the effectiveness of our proposed better initial estimate, we show the results of ATGV method being initialized with the bicubic interpolation method, which is referred to as ATGVBic here. MSE performance metric is used to evaluate the performance of the methods.

Figure 7.4 shows the SR results for various upsampling factor 2, 4 and 8 respectively on *noisy* depth input *Cones* with additive Gaussian noise of standard deviation $\sigma = 5$ which is the highest noise considered in the experiments. As one can see in Figure 7.4, the output of the proposed method (last row *ATGVMod*) are more sharper than all other comparative methods. First column shows the SR results for upsampling factor 2, and the subsequent columns (columns-2 and columns-3) shows the SR results for upsampling factor 4 and 8 respectively. The details in ATGV Mod is not much distinguishable in $\times 2$ upsampling case as compared to other methods, but it can be noticed carefully that the head and the stick region in the output image produced by ATGV Mod are sharper and noise free as compared to other methods. As the upsampling factor goes higher, the distinction between the output produced by the proposed method ATGV Mod (shown in last row) is more clear with sharper edge discontinuities.

In yet another example image of *Art*, whose qualitative results are shown in Figure 7.5, one can see that the output produced by the proposed method ATGV Mod are more sharper for all the upsampling factors.

For better visual representation, a small region cropped and zoomed from the SR outputs produced by the proposed method under different noise levels of $\sigma = 1, 2, 3, 4$ and 5, and for different upsampling factors of $\times 2$, $\times 4$ and $\times 8$ are shown in Figure 7.6, Figure 7.7 and Figure 7.8 respectively. It can be noticed from the cropped region (*teddy head*) clearly that the proposed method is able to suppress the noise to a larger extent and is able to maintain the depth precision and edge discontinuities in the generated SR output.

Table 7.1 and Table 7.2 shows the MSE performance metric of SR methods on few selected test depth images taken from Middlebury dataset (Scharstein and Szeliski, 2003). The average MSE results are shown for different noise levels (i.e. noise standard deviation 1, 2, 3, 4 and 5) and for different upsampling factors (i.e. 2, 4, and 8). Overall the proposed guidance based depth image SR method performs better than other competitive methods for various upsampling factor under different levels of noise.

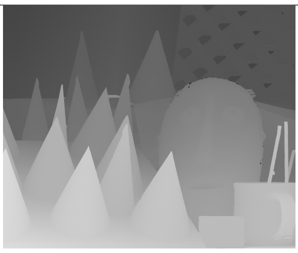
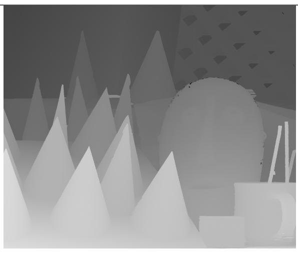
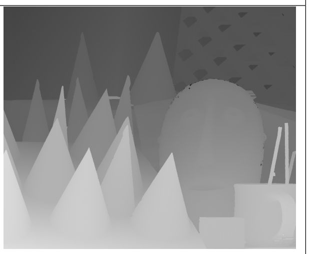
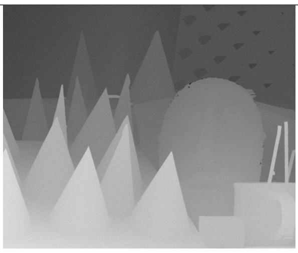
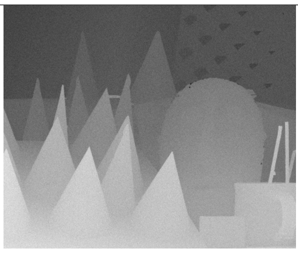
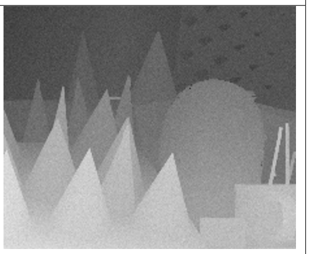
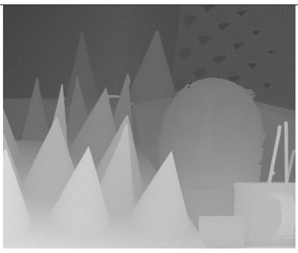
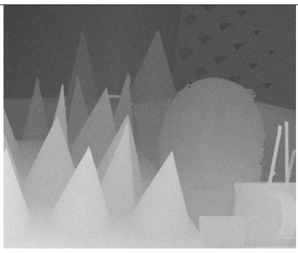
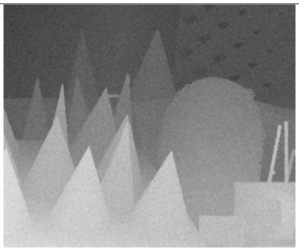
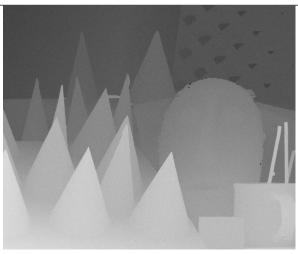
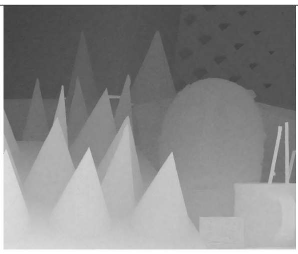
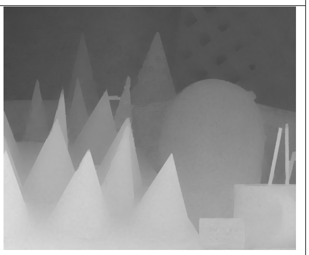
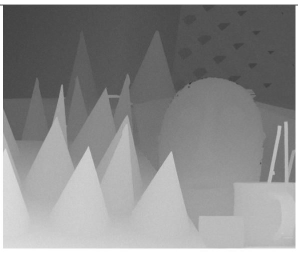
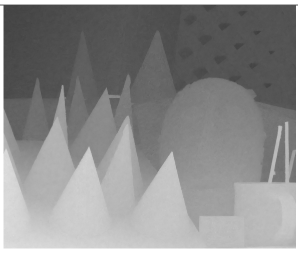
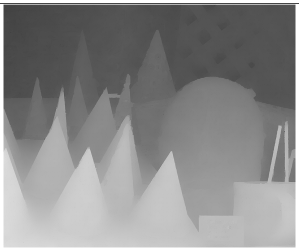
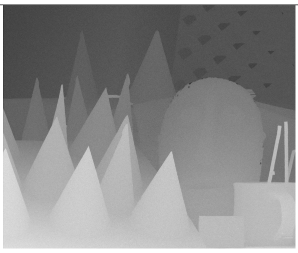
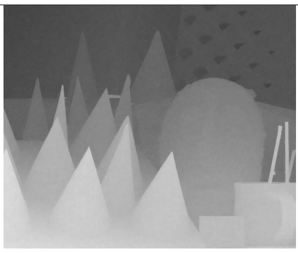
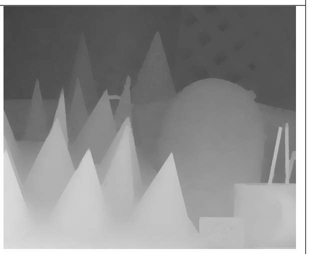
Factor	$\times 2$	$\times 4$	$\times 8$
GT			
Bic			
RI			
ATGV			
ATG Bic			
ATGV Mod			

Figure 7.4: SR results comparison of *noisy* ($\sigma = 5$) depth image *Cones*. **Col1**: SR by factor $\times 2$, **Col2**: SR by factor $\times 4$, **Col3**: SR by factor $\times 8$.

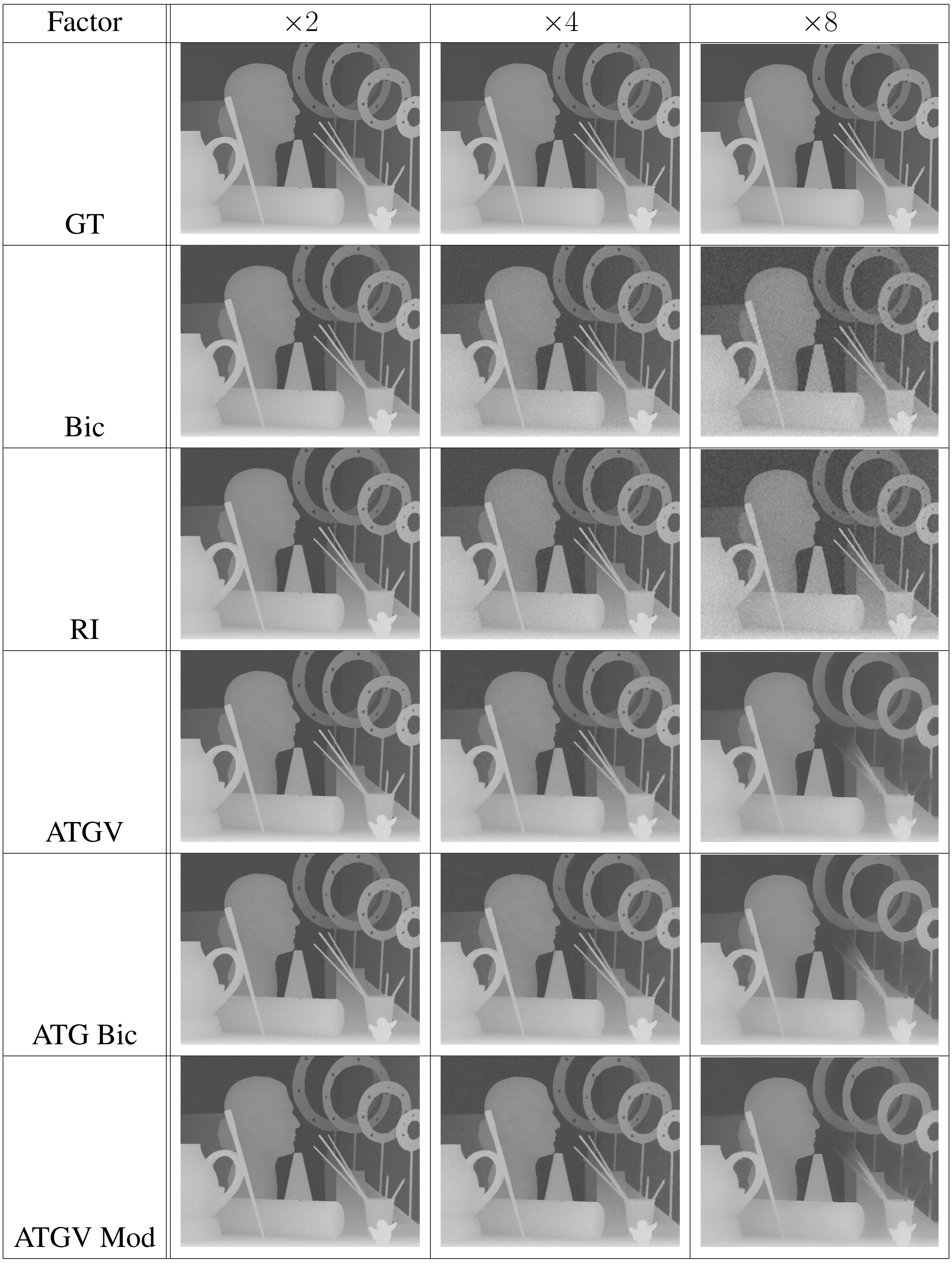
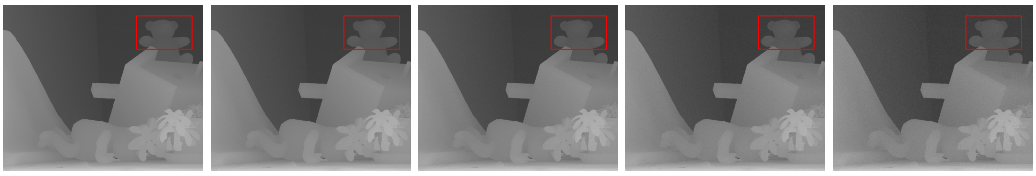
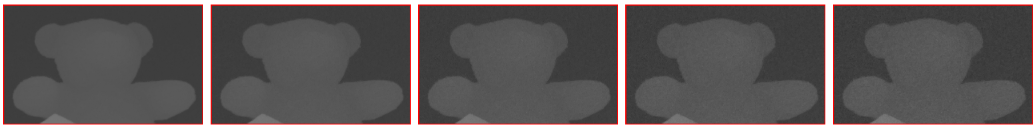


Figure 7.5: SR results comparison of *noisy* ($\sigma = 5$) depth image *Art*. **Col1**: SR by factor $\times 2$, **Col2**: SR by factor $\times 4$, **Col3**: SR by factor $\times 8$.

Table 7.1 shows MSE results on image with lowest ($\sigma = 1$) and highest ($\sigma = 5$) noise level. It shows that overall the proposed cascade method performs well, especially for higher upsampling factor, which is more important in case of depth image super-

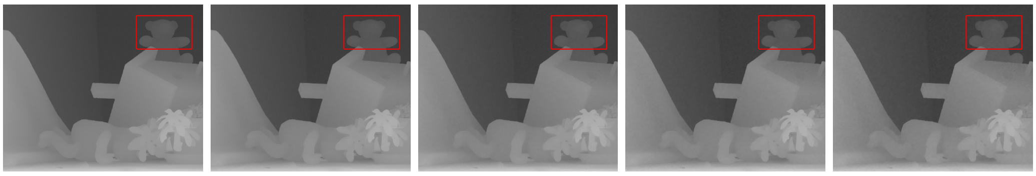


(a) ATGV Mod ($\sigma 1$) (b) ATGV Mod ($\sigma 2$) (c) ATGV Mod ($\sigma 3$) (d) ATGV Mod ($\sigma 4$) (e) ATGV Mod ($\sigma 5$)

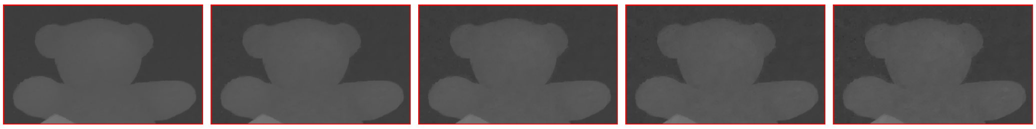


(f) (g) (h) (i) (j)

Figure 7.6: SR results of the proposed method under different noise levels for upsampling factors $\times 2$. **Row1**: outputs from proposed method, **Row2**: cropped and zoomed region of their corresponding top images.



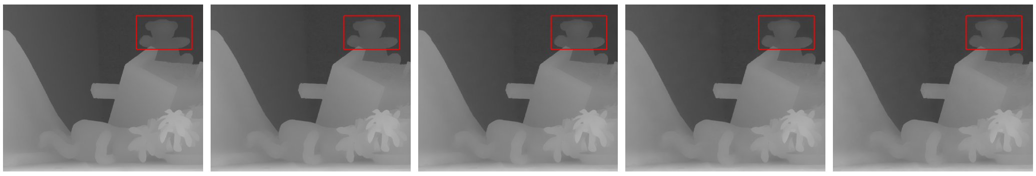
(a) ATGV Mod ($\sigma 1$) (b) ATGV Mod ($\sigma 2$) (c) ATGV Mod ($\sigma 3$) (d) ATGV Mod ($\sigma 4$) (e) ATGV Mod ($\sigma 5$)



(f) (g) (h) (i) (j)

Figure 7.7: SR results of the proposed method under different noise levels for upsampling factors $\times 4$. **Row1**: outputs from proposed method, **Row2**: cropped and zoomed region of their corresponding top images.

resolution. For lower level of noise, the proposed method performs comparatively better than other guidance based depth image SR methods. However, with higher level of noise, the proposed method is able to show good results for higher upsampling factors. Table 7.2 shows MSE results on same set of test images but with other levels of added noise, i.e. $\sigma = 2, 3$ and 4. This experiment was performed to see the performance of the proposed method under different level of noise. However, in this scenario also, the proposed method perform better than other SR methods for SR factors 4 and 8.



(a) ATGV Mod ($\sigma1$) (b) ATGV Mod ($\sigma2$) (c) ATGV Mod ($\sigma3$) (d) ATGV Mod ($\sigma4$) (e) ATGV Mod ($\sigma5$)

(f) (g) (h) (i) (j)

Figure 7.8: SR results of the proposed method under different noise levels for upsampling factors $\times8$. **Row1**: outputs from proposed method, **Row2**: cropped and zoomed region of their corresponding top images.

Table 7.1: MSE results of depth super-resolution by factor $\times2$, $\times4$ and $\times8$ for lowest and highest noise levels, i.e. $\sigma = 1$ and $\sigma = 5$

Images	Bic	RI	ATGV	ATGV Bic	ATGV Mod	Bic	RI	ATGV	ATGV Bic	ATGV Mod	Bic	RI	ATGV	ATGV Bic	ATGV Mod
	$\times2\_\sigma1$					$\times4\_\sigma1$					$\times8\_\sigma1$				
Aloe	3.92	3.82	**3.61**	3.62	3.62	4.25	3.81	**3.75**	3.75	3.76	5.19	**3.95**	4.40	4.42	4.45
Art	2.00	1.86	**1.60**	1.60	1.61	2.43	1.84	1.71	1.70	**1.70**	3.68	**2.01**	2.78	2.79	2.76
Baby	1.33	1.29	**1.01**	1.10	1.02	1.46	1.27	**0.99**	1.01	1.00	1.84	1.31	**1.22**	1.26	1.25
Books	2.23	2.20	1.91	1.93	**1.91**	2.37	2.22	1.98	1.96	**1.95**	2.80	2.31	2.35	2.32	**2.26**
Bowling	2.58	2.52	2.23	2.24	**2.22**	2.78	2.50	2.36	2.35	**2.34**	3.40	**2.59**	2.81	2.82	2.72
Cones	4.76	4.72	4.45	4.46	**4.45**	4.92	4.72	4.54	4.55	**4.52**	5.34	4.78	4.94	4.91	**4.71**
Moebius	2.16	2.13	1.84	1.85	**1.84**	2.31	2.14	1.89	1.90	**1.89**	2.74	**2.23**	2.31	2.32	2.29
Plastic	1.29	1.26	0.90	0.91	**0.89**	1.39	1.25	0.92	0.92	**0.90**	1.68	1.31	1.10	1.08	**1.01**
Reindeer	2.05	1.99	1.71	1.73	**1.71**	2.28	1.98	1.75	1.76	**1.75**	2.98	**2.08**	2.27	2.28	2.21
Teddy	4.26	4.23	3.96	3.98	**3.96**	4.39	4.24	**3.99**	4.02	4.00	4.73	4.31	4.23	4.25	**4.23**
Average	2.65	2.60	2.32	2.33	**2.32**	2.85	2.59	2.38	2.40	**2.38**	3.43	**2.68**	2.84	2.86	2.78
	$\times2\_\sigma5$					$\times4\_\sigma5$					$\times8\_\sigma5$				
Aloe	6.29	6.17	**5.42**	5.44	5.43	6.55	6.10	**4.80**	4.82	4.81	7.31	6.17	5.24	5.23	**5.20**
Art	4.40	4.25	**3.34**	3.36	3.35	4.75	4.16	2.59	2.60	**2.59**	5.78	4.27	3.61	3.61	**3.57**
Baby	3.84	3.76	**2.83**	2.84	2.84	3.94	3.69	1.84	1.85	**1.84**	4.23	3.70	1.87	1.86	**1.85**
Books	4.71	4.65	3.70	3.71	**3.70**	4.80	4.61	2.87	2.84	**2.83**	5.10	4.66	3.01	2.99	**2.90**
Bowling	5.04	4.95	3.95	3.98	**3.95**	5.18	4.88	3.18	3.18	**3.16**	5.69	4.93	3.57	3.56	**3.48**
Cones	7.16	7.08	**6.20**	6.22	6.21	7.29	7.04	5.41	5.42	**5.39**	7.59	7.05	5.50	5.48	**5.37**
Moebius	4.62	4.56	3.64	3.65	**3.64**	4.72	4.52	2.75	2.76	**2.75**	5.03	4.56	3.12	3.12	**3.08**
Plastic	3.81	3.75	2.65	2.68	**2.65**	3.88	3.69	1.65	1.64	**1.63**	4.10	3.73	1.65	1.64	**1.57**
Reindeer	4.52	4.42	3.50	3.51	**3.50**	4.70	4.37	2.59	2.60	**2.59**	5.26	4.44	2.97	2.96	**2.91**
Teddy	6.70	6.64	5.75	5.78	**5.75**	6.79	6.60	4.82	4.83	**4.82**	7.01	6.63	4.80	4.81	**4.76**
Average	5.10	5.02	**4.09**	4.10	4.10	5.26	4.96	3.25	3.26	**3.24**	5.71	5.01	3.53	3.55	**3.46**

7.5 SUMMARY

The proposed guidance colour image based depth image super-resolution method, which is a combination of residual interpolation method (RI) and anisotropic total generalized variation method (ATGV) in the SR pipeline in a single framework proves to be efficient in producing SR output under noisy conditions. With the strong intuition about

Table 7.2: MSE results of depth super-resolution by factor $\times 2$, $\times 4$ and $\times 8$ for different noise levels, i.e. $\sigma = 2, 3$ and 4

Images	Bic	RI	ATGV	ATGV Mod	Bic	RI	ATGV	ATGV Mod	Bic	RI	ATGV	ATGV Mod
	$\times 2\_\sigma 2$				$\times 4\_\sigma 2$				$\times 8\_\sigma 2$			
Aloe	4.50	4.40	**4.07**	4.08	4.80	4.38	**4.04**	4.05	5.68	4.49	**4.62**	4.63
Art	2.59	2.45	**2.05**	2.06	2.98	2.41	1.96	**1.95**	4.16	2.56	3.00	**2.98**
Baby	1.95	1.91	1.49	**1.49**	2.07	1.87	1.23	**1.23**	2.42	1.91	**1.40**	1.41
Books	2.84	2.81	2.37	**2.37**	2.96	2.81	2.23	**2.20**	3.33	2.89	2.51	**2.42**
Bowling	3.19	3.13	2.66	**2.66**	3.36	3.10	2.61	**2.59**	3.94	3.17	3.00	**2.92**
Cones	5.35	5.30	4.89	**4.89**	5.50	5.30	4.79	**4.77**	5.88	5.34	5.06	**4.89**
Moebius	2.77	2.73	**2.30**	2.31	2.89	2.73	2.14	**2.14**	3.28	2.80	2.53	**2.51**
Plastic	1.91	1.89	1.35	**1.35**	2.00	1.87	1.14	**1.11**	2.26	1.92	1.26	**1.16**
Reindeer	2.66	2.59	**2.16**	2.17	2.87	2.57	1.99	**1.98**	3.51	2.66	2.45	**2.40**
Teddy	4.86	4.83	4.42	**4.42**	4.97	4.83	4.23	**4.23**	5.27	4.89	4.38	**4.38**
Average	3.26	3.20	**2.77**	2.78	3.44	3.18	2.63	**2.62**	3.97	3.26	3.02	**2.97**
	$\times 2\_\sigma 3$				$\times 4\_\sigma 3$				$\times 8\_\sigma 3$			
Aloe	5.10	4.99	**4.52**	4.53	5.38	4.95	4.30	**4.30**	6.21	5.05	4.83	**4.83**
Art	3.19	3.05	2.49	**2.49**	3.56	2.99	2.17	**2.17**	4.68	3.12	3.21	**3.19**
Baby	2.58	2.52	1.94	**1.94**	2.69	2.48	1.44	**1.44**	3.01	2.50	1.56	**1.55**
Books	3.46	3.42	2.81	**2.81**	3.57	3.41	2.45	**2.41**	3.91	3.48	2.68	**2.58**
Bowling	3.80	3.74	3.09	**3.09**	3.97	3.69	2.81	**2.79**	4.51	3.76	3.21	**3.12**
Cones	5.95	5.90	5.33	**5.33**	6.09	5.88	5.00	**4.99**	6.44	5.91	5.22	**5.06**
Moebius	3.38	3.34	2.75	**2.75**	3.50	3.32	2.35	**2.35**	3.85	3.39	2.73	**2.71**
Plastic	2.55	2.51	1.79	**1.79**	2.63	2.48	1.32	**1.30**	2.87	2.53	1.39	**1.31**
Reindeer	3.28	3.20	2.61	**2.61**	3.47	3.16	2.19	**2.19**	4.08	3.26	2.63	**2.57**
Teddy	5.48	5.44	**4.86**	4.87	5.57	5.42	4.43	**4.43**	5.84	5.47	4.51	**4.51**
Average	3.87	3.81	**3.21**	3.22	4.04	3.77	2.84	**2.83**	4.54	3.84	3.19	**3.14**
	$\times 2\_\sigma 4$				$\times 4\_\sigma 4$				$\times 8\_\sigma 4$			
Aloe	5.70	5.58	**4.97**	4.98	5.97	5.52	**4.55**	4.56	6.75	5.61	5.04	**5.03**
Art	3.80	3.64	2.92	**2.92**	4.16	3.57	2.38	**2.38**	5.23	3.70	3.41	**3.38**
Baby	3.21	3.14	2.39	**2.39**	3.32	3.08	1.64	**1.64**	3.62	3.10	1.71	**1.70**
Books	4.09	4.04	3.26	**3.26**	4.18	4.01	2.66	**2.62**	4.50	4.07	2.84	**2.74**
Bowling	4.42	4.34	3.52	**3.52**	4.58	4.28	3.00	**2.98**	5.10	4.35	3.40	**3.31**
Cones	6.55	6.49	**5.76**	5.77	6.69	6.46	5.21	**5.19**	7.02	6.48	5.36	**5.22**
Moebius	4.00	3.95	**3.19**	3.20	4.11	3.92	2.56	**2.56**	4.44	3.97	2.93	**2.90**
Plastic	3.18	3.13	2.22	**2.22**	3.26	3.08	1.48	**1.46**	3.49	3.13	1.52	**1.44**
Reindeer	3.90	3.81	**3.05**	3.06	4.08	3.76	2.40	**2.39**	4.67	3.85	2.79	**2.74**
Teddy	6.09	6.04	5.31	**5.31**	6.18	6.01	4.63	**4.63**	6.42	6.05	4.66	**4.63**
Average	4.49	4.41	**3.65**	3.66	4.65	4.36	3.05	**3.04**	5.12	4.43	3.36	**3.30**

obtaining better output with a better initial HR estimate, the proposed method cascades RI and ATGV method, where RI module provides a better initial estimate, and the ATGV module improve the SR accuracy with faster convergence. The initial estimate produced by RI module is better because it is as fast as any other interpolation method. The qualitatively and quantitatively experimental results shows that the proposed performs comparative well for upsampling factor 2, however it performs better by maintaining the depth precision and the edge discontinuities as compared to other SR methods, especially for higher upsampling factor 4 and 8, which is a good sign.

CHAPTER 8

CONCLUSION AND FUTURE DIRECTIONS

8.1 CONCLUSION

With recent advancements in the image processing and computer vision field, the demand of depth images have increased. Several applications which demand depth images require high-resolution depth images, but the commercial depth cameras could not meet the demand of these applications. The images captured by these cameras suffer from lower spatial resolution, corrupted with noise, and have missing regions.

This thesis presented some methods for super-resolving the depth images from uniform samples and densely reconstructing the depth images from non-uniform samples. The proposed wavelet based SR method in this thesis is a simple and efficient method. It utilizes DWT, SWT and gradient of the input image to enhance the high-frequency content in the image. This is because the prominent features of depth images are the edges, and these edges are well captured in the subbands of DWT, SWT and gradient. Fusing these details has produced a better super-resolved image On the test images, the proposed methods performs better in terms of retaining the edge discontinuities and

The guidance image based SR method presented here use HR colour image as guidance image of the same scene as that of the observed depth images. The proposed SR method shows two variants based on the use of the LR input image. If the LR image is initially bicbic interpolated and fed to the proposed SR pipeline, here it is called as LRBicSR method, and if the LR image is mapped on the HR grid and fed as input to the proposed SR pipeline, then it is called as LRSR method. The proposed method utilize the segment cues from the guidance image to guide the super-resolution process. Here, each segment region obtained from the colour image is looked for its corresponding segment region in the depth image. Based on some threshold value, it decides whether the regions is a smooth region or a probable edge region. Then it takes the decision to whether fill that depth segment with the bicubic values or with the median value. The proposed method has been demonstrated for higher upsampling factors also. Both the SR variants perform better than other comparative methods.

This thesis proposed a method for densely reconstructing a depth image from depth image with random sparse depth points. The depth points spread uniformly on the image. The proposed depth reconstruction (DR) method uses two different approaches to estimate the unknown pixels. One is the plane fitting approach (PFit) and other is the median filling approach (MFill). The PFit approach existed in literature but it was for some different configuration and the sparseness percentage considered was around 25-30% visible pixels. In the proposed method, the performance of the depth reconstruction is shown for the lowest sparseness of 1%. Reconstructing the complete image from just 1% random non-uniform depth points is really challenging. The proposed method utilize the guidance colour image to perform dense depth reconstruction. Other than depth reconstruction, this thesis also presents a method to super-resolve a non-uniform LR image, called LRSR method. It is the combination of DR and SR method in a single framework. The performance of the DRSR method is also better than the other comparative SR methods. It is also shown that the proposed method can be utilized to address the other depth image related problems like depth denoising and depth inpainting. Here, it is only shown the adaptability of the proposed method for depth denoising and depth inpainting gives better results, but it does not show the improvements in the results compared to other state-of-the-art denoising and inpainting methods.

The use of training images for super-resolving the depth images is also demonstrated in this thesis. The use of GMM proves to be efficient in doing the super-resolution task by learning the HR-LR relationship. This approach has been shown to super-resolve the images by higher SR factors. It is shown that, for higher SR factors, the direct approach may not achieve better results, so a hierarchical approach has been demonstrated which performs the super-resolution in steps of $\times 2$ to achieve higher SR factor. The experimentation has been done by varying several parameters, i.e. different HR-LR patch sizes and training different number of GMMs. We have seen that GMM with 200-250 gave good results.

For iterative methods, this thesis proposed a cascade approach for better initial estimate. It combines the residual interpolation (RI) method with anisotropic total generalized variation (ATGV). A simple and less computationally intensive Residual interpolation method (RI) has been used as a preprocessor for ATGV. RI method is very less computationally intensity, and it can be very well compared with bicubic interpolation method in terms of computation time. It is observed that the proposal of cascading the RI as a preprocessor reduces the number of iterations, converges faster to achieve the better SR image quality.

8.2 FUTURE DIRECTIONS

The current state-of-the-art results show improved performance of the super-resolution methods. However, there is always some scope to improve the results and perform better than the existing methods.

From my point of view, one of the future work based on this thesis could be to increase the computational speed the super-resolution methods. The time taken by the existing super-resolution methods are in few seconds, which makes it unreliable for video super-resolution. With recent camera revolution, there might be a need of depth video super-resolution where the per frame computation has to match in accordance with the depth video frame rates. The application could be the autonomous driving vehicle where the vehicle has to monitor the current situation on the road by continuously monitoring the depth video for any abnormalities.

Over the past few years, the employment of convolutional neural network (CNN) has seen a rapid growth in many fields of image processing and computer vision. Recently, there has been a rapid increase in the use of CNN form optical image super-resolution. A similar approach of CNN and related deep learning techniques can be used for depth image super-resolution. However, the direct utilization of the existing nets for depth image super-resolution is non-trivial. The depth image properties have to be understood properly, and then the appropriate features can be learned by the CNN layers which might be useful for the depth image super-resolution task. The training process can be painful as it might require large dataset to learn the prominent features in the image. But, with recent advancement in graphical processing unit (GPU), the training time has drastically reduced.

The area of 3D depth reconstruction is not much explored. The existing methods work on 2D non-uniform samples where the samples are randomly placed on the image grid, and the depth reconstruction method tried to densely reconstruct the image by estimating the unknown pixels. A similar model can be build for 3D depth reconstruction. It can be used in several applications like augmented reality or in the medical field to know the complete anatomy of any body part.

Other than optical images and the depth images, the super-resolution methods can be applied on other modalities of images. For example, the medical images from MRI or CT are mostly corrupted by noise and have poor quality. The super-resolution methods can be implemented to resolve such problems. There are some existing SR methods for MRI images which produces good quality noise free super-resolved images. However, there is still a huge demand for providing the clearer images of other modalities.

APPENDIX A

INTERPOLATION METHODS

Nearest Neighbor Interpolation

In nearest neighbor (NN) interpolation the intermediate pixel value of SR frame is chosen to be the nearest among the neighbors. This technique give rise to the blocking artifacts on the edges and results in unpleasing SR image. As shown in the Figure A.1, the interpolating data points will be assigned the value amongst the neighbor to which it is closer/nearer.

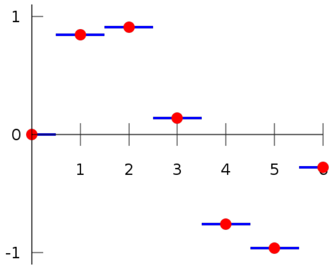


Figure A.1: Nearest Neighbor Interpolation

Bilinear Interpolation

This technique is the extension of linear interpolation function. Figure A.2 shows how bilinear interpolation is performed. The *red* dots are the available pixels from the LR image, and the *green* dot at the center is the estimation point. The linear interpolation can be applied twice on 2D image, once in one direction and then in another direction, as bilinear operation is separable. The *blue* dots are the result of linear interpolation in x-direction, and the *green* dot are the result of linear interpolation in y-direction on the points obtained in previous step. The computation of theses unknown points from the known points is shown in the Eq. A.1.

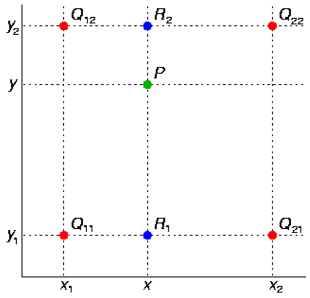


Figure A.2: Bilinear Interpolation

$$
\begin{aligned}
f(R_1) &= \frac{x_2 - x}{x_2 - x_1} f(Q_{11}) + \frac{x - x_1}{x_2 - x_1} f(Q_{21}) \\
f(R_2) &= \frac{x_2 - x}{x_2 - x_1} f(Q_{12}) + \frac{x - x_1}{x_2 - x_1} f(Q_{22}) \\
f(R_1) &= \frac{y_2 - y}{y_2 - y_1} f(R_1) + \frac{y - y_1}{y_2 - y_1} f(R_2)
\end{aligned}
\tag{A.1}
$$

Spline Interpolation

Spline interpolation techniques (Lalescu, 2009) use low degree polynomials for each intervals, and chooses the polynomial pieces such that they fit smoothly together. Bicubic Interpolation Technique (Keys, 1981) is generally preferred over bilinear or nearest neighbor interpolation. The reason is that the interpolated points follow the smooth transition and it has fewer interpolation artifacts. The only drawback of this technique is that it is computationally demanding. Keys (1981) has derived cubic convolution interpolation for 1D, and the extension of this algorithm to 2D is applied to 2D image data as shown in Figure A.3.

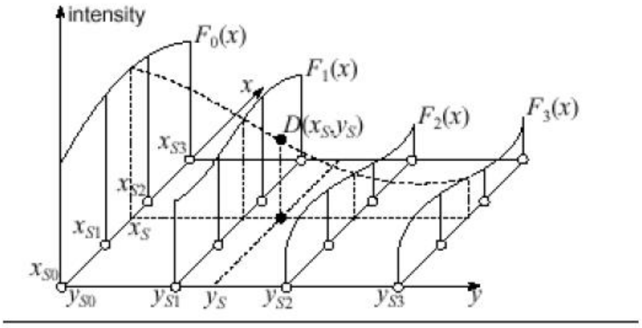


Figure A.3: Bicubic Interpolation

Polynomial Interpolation

It is the generalized linear interpolation technique. In this interpolation method, one uses higher order polynomial which goes through all the give set of discrete points. Since it is infinitely differentiable, it overcomes so many problems of linear interpolation. The disadvantage of such method is that it is very computationally intensive, and it may contain oscillatory artifacts.

Lanczos Interpolation

It is also called as *Lanczos resampling* or *Lanczos filter*. It is a mathematical formula used to smoothly interpolate the values of the digital signal between its samples. It maps each sample of the given signal to a translated and scaled copy of the Lanczos kernel which is a *sinc* function windowed by the central hump of a dialated sinc function. The sum of these translated and scaled kernels is then evaluated at the desired point, which is given by Eq. A.2. The Lanczos kernel for a=3 is shown in Figure A.4.

$$\begin{aligned} f(n) &= sinc(x)\ sinc(x/a); & if\ -a < x < a \\ &= 0; & otherwise \end{aligned} \tag{A.2}$$

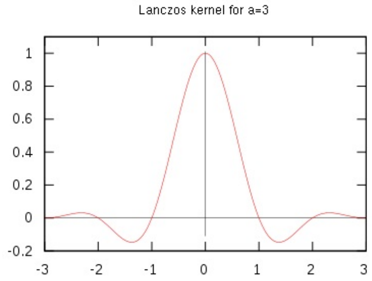


Figure A.4: Lanczos kernel for a=3

REFERENCES

Achanta, Radhakrishna, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk (2012), "Slic superpixels compared to state-of-the-art superpixel methods." *IEEE transactions on pattern analysis and machine intelligence*, 34, 2274–2282.

Bhattacharya, Saumik, Sumana Gupta, and KS Venkatesh (2014), "High accuracy depth filtering for kinect using edge guided inpainting." *Advances in Computing, Communications and Informatics (ICACCI, 2014 International Conference on*, 868–874, IEEE.

Bhavsar, Arnav V and Ambasamudram N Rajagopalan (2012), "Range map superresolution-inpainting, and reconstruction from sparse data." *Computer Vision and Image Understanding*, 116, 572–591.

Chan, Derek, Hylke Buisman, Christian Theobalt, and Sebastian Thrun (2008), "A noise-aware filter for real-time depth upsampling." *Workshop on Multi-camera and Multi-modal Sensor Fusion Algorithms and Applications-M2SFA2 2008*.

Comaniciu, Dorin and Peter Meer (2002), "Mean shift: A robust approach toward feature space analysis." *IEEE Transactions on pattern analysis and machine intelligence*, 24, 603–619.

Demirel, Hasan and Gholamreza Anbarjafari (2010), "Satellite image resolution enhancement using complex wavelet transform." *IEEE geoscience and remote sensing letters*, 7, 123–126.

Demirel, Hasan and Gholamreza Anbarjafari (2011a), "Discrete wavelet transform-based satellite image resolution enhancement." *IEEE transactions on geoscience and remote sensing*, 49, 1997–2004.

Demirel, Hasan and Gholamreza Anbarjafari (2011b), “Image resolution enhancement by using discrete and stationary wavelet decomposition.” *IEEE transactions on image processing*, 20, 1458–1460.

Diebel, James and Sebastian Thrun (2005), “An application of markov random fields to range sensing.” *NIPS*, 5, 291–298.

Dong, Chao, Chen Change Loy, Kaiming He, and Xiaoou Tang (2016), “Image super-resolution using deep convolutional networks.” *IEEE transactions on pattern analysis and machine intelligence*, 38, 295–307.

Ferstl, David, Christian Reinbacher, Rene Ranftl, Matthias Rüther, and Horst Bischof (2013), “Image guided depth upsampling using anisotropic total generalized variation.” *Proceedings of the IEEE International Conference on Computer Vision*, 993–1000.

Fischler, Martin A and Robert C Bolles (1981), “Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography.” *Communications of the ACM*, 24, 381–395.

Freeman, William T, Thouis R Jones, and Egon C Pasztor (2002), “Example-based super-resolution.” *IEEE Computer graphics and Applications*, 22, 56–65.

Garcia, Frederic, Djamila Aouada, Bruno Mirbach, Thomas Solignac, and Björn Ottersten (2011), “A new multi-lateral filter for real-time depth enhancement.” *Advanced Video and Signal-Based Surveillance (AVSS), 2011 8th IEEE International Conference on*, 42–47, IEEE.

Garcia, Frederic, Djamila Aouada, Bruno Mirbach, Thomas Solignac, and Björn Ottersten (2015), “Unified multi-lateral filter for real-time depth map enhancement.” *Image and Vision Computing*, 41, 26–41.

Garcia, Frederic, Bruno Mirbach, Bjorn Ottersten, Frédéric Grandidier, and Angel Cuesta (2010), “Pixel weighted average strategy for depth sensor data fusion.” *Image Processing (ICIP), 2010 17th IEEE International Conference on*, 2805–2808, IEEE.

Gevrekci, Murat and Kubilay Pakin (2011), “Depth map super resolution.” *Image Processing (ICIP), 2011 18th IEEE International Conference on*, 3449–3452, IEEE.

Glasner, Daniel, Shai Bagon, and Michal Irani (2009), “Super-resolution from a single image.” *Computer Vision, 2009 IEEE 12th International Conference on*, 349–356, IEEE.

Gupta, Lalit and Thotsapon Sortrakul (1998), “A gaussian-mixture-based image segmentation algorithm.” *Pattern Recognition*, 31, 315–325.

Ham, Bumsub, Dongbo Min, and Kwanghoon Sohn (2015), “Depth superresolution by transduction.” *IEEE Transactions on Image Processing*, 24, 1524–1535.

He, Kaiming, Jian Sun, and Xiaoou Tang (2010), “Guided image filtering.” *European conference on computer vision*, 1–14, Springer.

Herrera, Daniel, Juho Kannala, Janne Heikkilä, et al. (2013), “Depth map inpainting under a second-order smoothness prior.” *Scandinavian Conference on Image Analysis*, 555–566, Springer.

Hua, Kai-Lung, Kai-Han Lo, and Yu-Chiang Frank Frank Wang (2016), “Extended guided filtering for depth map upsampling.” *IEEE MultiMedia*, 23, 72–83.

Huang, Jia-Bin, Abhishek Singh, and Narendra Ahuja (2015), “Single image super-resolution from transformed self-exemplars.” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5197–5206.

Jiji, CV, Manjunath V Joshi, and Subhasis Chaudhuri (2004), “Single-frame image super-resolution using learned wavelet coefficients.” *International journal of Imaging systems and Technology*, 14, 105–112.

Keys, Robert (1981), “Cubic convolution interpolation for digital image processing.” *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 29, 1153–1160.

Kil, Yong Joo, Boris Mederos, and Nina Amenta (2006), “Laser scanner super-resolution.” *SPBG*, 9–15.

Kim, Jiwon, Jung Kwon Lee, and Kyoung Mu Lee (2016), “Deeply-recursive convolutional network for image super-resolution.” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1637–1645.

Kim, Kwang In and Younghee Kwon (2008), “Example-based learning for single-image super-resolution.” *Joint Pattern Recognition Symposium*, 456–465, Springer.

Kim, Sung-Yeol, Ji-Ho Cho, Andreas Koschan, and Mongi A Abidi (2010), “Spatial and temporal enhancement of depth images captured by a time-of-flight depth sensor.” *Pattern Recognition (ICPR), 2010 20th International Conference on*, 2358–2361, IEEE.

Konno, Yosuke, Yusuke Monno, Daisuke Kiku, Masayuki Tanaka, and Masatoshi Okutomi (2015), “Intensity guided depth upsampling by residual interpolation.” *The... international conference on advanced mechatronics: toward evolutionary fusion of IT and mechatronics: ICAM: abstracts*, 2015, 1–2, .

Kopf, Johannes, Michael F Cohen, Dani Lischinski, and Matt Uyttendaele (2007), “Joint bilateral upsampling.” *ACM Transactions on Graphics (ToG)*, 26, 96, ACM.

Lalescu, Cristian Constantin (2009), “Two hierarchies of spline interpolations. practical algorithms for multivariate higher order splines.” *arXiv preprint arXiv:0905.3564*.

Li, Feng, Jingyi Yu, and Jinxiang Chai (2008), “A hybrid camera for motion deblurring and depth map super-resolution.” *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, 1–8, IEEE.

Li, Jing, Zhichao Lu, Gang Zeng, Rui Gan, and Hongbin Zha (2014), “Similarity-aware patchwork assembly for depth image super-resolution.” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3374–3381.

Li, Xin and Michael T Orchard (2001), “New edge-directed interpolation.” *IEEE transactions on image processing*, 10, 1521–1527.

Lim, Bee, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee (2017), “Enhanced deep residual networks for single image super-resolution.” *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.

Liu, Junyi, Xiaojin Gong, and Jilin Liu (2012), “Guided inpainting and filtering for kinect depth maps.” *Pattern Recognition (ICPR), 2012 21st International Conference on*, 2055–2058, IEEE.

Liu, Lee-Kang, Stanley H Chan, and Truong Q Nguyen (2015), “Depth reconstruction from sparse samples: Representation, algorithm, and sampling.” *IEEE Transactions on Image Processing*, 24, 1983–1996.

Liu, Ming-Yu, Oncel Tuzel, and Yuichi Taguchi (2013), “Joint geodesic upsampling of depth images.” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 169–176.

Lu, Jiajun and David Forsyth (2015), “Sparse depth super resolution.” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2245–2253.

Lucas, Bruce D, Takeo Kanade, et al. (1981), “An iterative image registration technique with an application to stereo vision.”

Luo, Wenjie, Alexander G Schwing, and Raquel Urtasun (2016), “Efficient deep learning for stereo matching.” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5695–5703.

Mac Aodha, Oisin, Neill DF Campbell, Arun Nair, and Gabriel J Brostow (2012), “Patch based synthesis for single depth image super-resolution.” *European Conference on Computer Vision*, 71–84, Springer.

Mallat, Stéphane (1999), *A wavelet tour of signal processing*. Academic press.

Mandal, Srimanta, Arnav Bhavsar, and Anil Kumar Sao (2017), “Depth map restoration from undersampled data.” *IEEE Transactions on Image Processing*, 26, 119–134.

Moon, Young Shik et al. (2015), “Super-resolution image reconstruction using wavelet based patch and discrete wavelet transform.” *Journal of Signal Processing Systems*, 81, 71–81.

Park, Jaesik, Hyeongwoo Kim, Yu-Wing Tai, Michael S Brown, and In So Kweon (2014), "High-quality depth map upsampling and completion for rgb-d cameras." *IEEE Transactions on Image Processing*, 23, 5559–5572.

Portilla, Javier, Vasily Strela, Martin J Wainwright, and Eero P Simoncelli (2003), "Image denoising using scale mixtures of gaussians in the wavelet domain." *IEEE Transactions on Image processing*, 12, 1338–1351.

Qi, Fei, Junyu Han, Pengjin Wang, Guangming Shi, and Fu Li (2013), "Structure guided fusion for depth map inpainting." *Pattern Recognition Letters*, 34, 70–76.

Reynolds, Douglas A, Thomas F Quatieri, and Robert B Dunn (2000), "Speaker verification using adapted gaussian mixture models." *Digital signal processing*, 10, 19–41.

Sandeep, Palakkattillam and Tony Jacob (2016), "Single image super-resolution using a joint gmm method." *IEEE Transactions on Image Processing*, 25, 4233–4244.

Scharstein, Daniel and Richard Szeliski (2002), "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms." *International journal of computer vision*, 47, 7–42.

Scharstein, Daniel and Richard Szeliski (2003), "High-accuracy stereo depth maps using structured light." *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, 1, I–195, IEEE.

Schuon, Sebastian, Christian Theobalt, James Davis, and Sebastian Thrun (2008), "High-quality scanning using time-of-flight depth superresolution." *Computer Vision and Pattern Recognition Workshops, 2008. CVPRW'08. IEEE Computer Society Conference on*, 1–7, IEEE.

Song, Xibin, Yuchao Dai, and Xueying Qin (2016), "Deep depth super-resolution: Learning depth super-resolution using deep convolutional neural network." *Asian Conference on Computer Vision*, 360–376, Springer.

Sun, Jian, Zongben Xu, and Heung-Yeung Shum (2008), “Image super-resolution using gradient profile prior.” *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, 1–8, IEEE.

Sun, Jian, Zongben Xu, and Heung-Yeung Shum (2011), “Gradient profile prior and its applications in image super-resolution and enhancement.” *IEEE Transactions on Image Processing*, 20, 1529–1542.

Temizel, Alptekin, Theo Vlachos, and W Visioprime (2005), “Wavelet domain image resolution enhancement using cycle-spinning.” *Electronics Letters*, 41, 119–121.

Tomasi, Carlo and Roberto Manduchi (1998), “Bilateral filtering for gray and color images.” *Computer Vision, 1998. Sixth International Conference on*, 839–846, IEEE.

Wang, Zhou, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli (2004), “Image quality assessment: from error visibility to structural similarity.” *IEEE transactions on image processing*, 13, 600–612.

Xiao, Lei, Felix Heide, Matthew O’Toole, Andreas Kolb, Matthias B Hullin, Kyros Kutulakos, and Wolfgang Heidrich (2015), “Defocus deblurring and superresolution for time-of-flight depth cameras.” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2376–2384.

Xie, Jun, Rogerio Schmidt Feris, and Ming-Ting Sun (2016), “Edge-guided single depth image super resolution.” *IEEE Transactions on Image Processing*, 25, 428–438.

Yadav, Mohit, Ram Garg, and Arnav Bhavsar (2014), “Better guiding the guided range image filter for range-image super-resolution.” *Proceedings of the 2014 Indian Conference on Computer Vision Graphics and Image Processing*, 62, ACM.

Yang, Jianchao, John Wright, Thomas S Huang, and Yi Ma (2010), “Image super-resolution via sparse representation.” *IEEE transactions on image processing*, 19, 2861–2873.

Yang, Jingyu, Xinchen Ye, Kun Li, Chunping Hou, and Yao Wang (2014), "Color-guided depth recovery from RGB-D data using an adaptive autoregressive model." *IEEE Transactions on Image Processing*, 23, 3443–3458.

Yang, Qingxiong (2012), "A non-local cost aggregation method for stereo matching." *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, 1402–1409, IEEE.

Yang, Qingxiong, Narendra Ahuja, Ruigang Yang, Kar-Han Tan, James Davis, Bruce Culbertson, John Apostolopoulos, and Gang Wang (2013), "Fusion of median and bilateral filtering for range image upsampling." *IEEE Transactions on Image Processing*, 22, 4841–4852.

Yang, Qingxiong, Ruigang Yang, James Davis, and David Nistér (2007), "Spatial-depth super resolution for range images." *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, 1–8, IEEE.

Yang, Yuxiang and Zengfu Wang (2012), "Range image super-resolution via guided image filter." *Proceedings of the 4th International Conference on Internet Multimedia Computing and Service*, 200–203, ACM.

Yin, Haitao, Shutao Li, and Leyuan Fang (2013), "Simultaneous image fusion and super-resolution using sparse representation." *Information Fusion*, 14, 229–240.

Zivkovic, Zoran (2004), "Improved adaptive gaussian mixture model for background subtraction." *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, 2, 28–31, IEEE.

LIST OF PAPERS BASED ON THESIS

Referred International Journals

1. **Chandra Shaker Balure**, Arnav Bhavsar, and M. Ramesh Kini . ”Guided Depth Image Reconstruction From Very Sparse Measurements.” *SPIE, Journal of Electronic Imaging.*
   DOI: 10.1117/1.JEI.27.5.053016
2. **Chandra Shaker Balure**, M. Ramesh Kini . ”Guidance Based Improved Depth Upsampling With Better Initial Estimate.” *Inderscience, International Journal of Computational Vision and Robotics.* **[Under Review]**

Referred National/International Conference Proceedings

1. **Chandra Shaker Balure**, and M. Ramesh Kini. ”A Survey–Super Resolution Techniques for Multiple, Single, and Stereo Images.” *Fifth International Symposium on Electronic System Design (ISED-2014).* IEEE, 2014.
   DOI: 10.1109/ISED.2014.53, Electronic ISBN: 978-1-4799-6965-4.
2. **Chandra Shaker Balure**, and M. Ramesh Kini. ”Depth Image Super Resolution - A Review and Wavelet Perspective.” *Computer Vision and Image Processing (CVIP).*
   DOI: 10.1007/978-981-10-2107-7_49, Online ISBN: 978-981-10-2107-7.
3. **Chandra Shaker Balure**, M. Ramesh Kini, and Arnav Bhavsar. ”Single Depth Image Super-Resolution via High-Frequency Subbands Enhancement and Bilateral Filtering.” *Eleventh International Conference on Industrial and Information Systems (ICIIS-2016)* IEEE, 2016.
   DOI: 10.1109/ICIINFS.2016.8262996, Electronic ISBN: 978-1-5090-3818-3.
4. **Chandra Shaker Balure**, M. Ramesh Kini, and Arnav Bhavsar. ”Depth Image Super-resolution with Local Medians and Bilateral Filtering.” *Eleventh International Conference on Industrial and Information Systems (ICIIS-2016)* IEEE, 2016.
   DOI: 10.1109/ICIINFS.2016.8263062, Electronic ISBN: 978-1-5090-3818-3.
5. **Chandra Shaker Balure**, Arnav Bhavsar, and M. Ramesh Kini. ”Local Segment-Based Dense Depth Reconstruction from Very Sparsely Sampled Data.” *National Conference on Communications (NCC-2017)* IEEE, 2017.
   DOI: 10.1109/NCC.2017.8077134, Electronic ISBN: 978-1-5090-5356-8.

6. **Chandra Shaker Balure**, Arnav Bhavsar, and M. Ramesh Kini. ”GMM Based Depth Image Super-Resolution.” *National Conference on Computer Vision, Pattern Recognition, Image Processing and Graphics (NCVPRIPG 2017).* **DOI**: 10.1007/978-981-13-0020-2_22

CURRICULUM VITAE

Name	Chandra Shaker Balure
Address	Mahadevi Nilay, H# 19-4-641/2, Naubad, Bidar - 585402, Karnataka, India.
E-mail	balure_1986a@yahoo.co.in | balure1986a@gmail.com
Qualification	• M.Tech. | VLSI Design and Embedded System | VTU University Karnataka • B.E. | Electronics and Telecommunication | S.R.T.M.U. Nanded Maharastra
Experience	• Lecturer | New Horizon College of Engineering, Bangalore • Project Trainee | LSI R& D, Bangalore • Member Technical Staff | HCL Technologies, Bangalore • Faculty | Gopalan College of Engineering and Management, Bangalore • Assistant Professor | SVERI's College of Engineering, Pandharpur