Faculty of Engineering and Information Technology
University of Technology Sydney

# High-Quality Depth Maps Acquisition for RGB-D Data

A thesis submitted in partial fulfillment of
the requirements for the degree of
**Doctor of Philosophy**

by

Yifan Zuo

April 2018

## CERTIFICATE OF AUTHORSHIP/ORIGINALITY

I certify that the work in this thesis has not previously been submitted for a degree nor has it been submitted as part of requirements for a degree except as fully acknowledged within the text.

I also certify that the thesis has been written by me. Any help that I have received in my research work and the preparation of the thesis itself has been acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

This thesis is the result of me conducted jointly with Shanghai University as part of a collaborative Doctoral degree.

Signature of Candidate: _____

Date: _____

# Acknowledgments

For two years in UTS which is a part of dual-doctoral plan, my deepest gratitude goes first and foremost to Dr. Qiang Wu, my supervisor, for the constant encouragement and guidance for my Ph.D work. Through weekly meetings, his immense knowledge and motivation help me address problems in research as well as writing this thesis efficiently.

I would like to express my heartfelt gratitude to my co-supervisor Associate Prof. Jian Zhan and my domestic supervisor Prof. Ping An in Shanghai University for providing me with continuous support throughout my PhD study and research.

I thank my labmates in Global Big Data Technology Center: Xiaoshui Huang, Junjie Zhang, Lina Li, Peng Zhang and Zongjian Zhang and labmates in Shanghai University: Xiwu Shang, Chao Yang, Deyang Liu, Jian Ma and Jianxin Wang for the help and discussions and for all the fun we have had in my PhD study period.

Last but not the least, my thanks would go to my parents for their support and great confidence in me all through these years.

Yifan Zuo
September 2017 @ UTS

# Contents

# List of Figures

# List of Tables

# List of Publications

**Papers Published (The First Author)**

- **Yifan Zuo**, Qiang Wu, Jian Zhang, Ping An (2018), Minimum Spanning Forest with Embedded Edge Inconsistency Measurement for Guided Depth Map Enhancement. *IEEE trans. IP*, published online.
- **Yifan Zuo**, Qiang Wu, Jian Zhang, Ping An (2018), Explicit Edge Inconsistency Evaluation Model for Color-guided Depth Map Enhancement. *IEEE trans. CSVT*, vol. 28, no. 2, pp. 439-453.
- **Yifan Zuo**, Qiang Wu, Jian Zhang, Ping An (2016), Explicit Modeling on Depth-Color Inconsistency for Color-Guided Depth Up-sampling. *in* 'Proceedings of the International conference on Multimedia and Expo (ICME16)', IEEE, pp. 1-6.
- **Yifan Zuo**, Qiang Wu, Jian Zhang, Ping An (2016), Explicit Measurement on Depth-Color Inconsistency for Depth Completion. *in* 'Proceedings of the International conference on image processing (ICIP16)', IEEE, pp. 4037-4041.
- **Yifan Zuo**, Ping An, Shuai Zheng, Zhaoyang Zhang (2015), Depth Upsampling Method via Markov Random Fields without Edge-Misaligned Artifacts, *in* 'Proceedings of the International conference on image processing (ICIP15)', IEEE, pp. 2324-2328.
- **Yifan Zuo**, Qiang Wu, Jian Zhang, Ping An (2017), Minimum Spanning Forest with Embedded Edge Inconsistency Measurement for Color-

guided Depth Map Upsampling, *in* 'Proceedings of the International conference on Multimedia and Expo (ICME17)', IEEE, pp. 211-216.

- **Yifan Zuo**, Ping An, Liquan Shen, Chunhua Li, Ran Ma (2016), Integration of Color and Affine Invariant Feature for Multi-view Depth Video Estimation. *Imaging Science Journal*, vol. 64, no. 6, pp. 313-320.

- **Yifan Zuo**, Ping An, Ran Ma, Liquan Shen, Zhaoyang Zhang (2014), Temporal Consistency Enhancement on Depth Sequences. *Journal of Optoelectronics Laser*, vol.25, no.1, pp.172-177.

- **Yifan Zuo**, Ping An, Qiu-Wen Zhang, and Zhao-Yang Zhang (2012), Fast Segment-Based Algorithm for Multi-view Depth Map Generation. *in* 'Proceedings of the International Conference on Intelligent Computing (ICIC12)', pp. 553-560.

**Papers to be Submitted/Under Review**

- **Yifan Zuo**, Qiang Wu, Ping An, Xiwu Shang, Integrate co-sparse analysis model with explicit edge inconsistency measurement for guided depth map upsampling. *Journal of Electronic Imaging*.

# Abstract

With the developing of computer vision, the problem of high-quality depth map acquisition is demanding urgent solution. Generally, the methods for dense depth map acquisition consist of two categories: passive and active.

The passive methods based on stereo matching algorithms always compute matching cost volume pixel by pixel, which is time-consuming. This thesis firstly proposes a local depth estimation method using adaptive matching scheme. Furthermore, with the help of affine invariant feature, the performance for matching in textureless regions is improved. Experimental results show that the proposed method can achieve better or comparable performances than the state-of-the-art method in the category of local methods, even with the less running time. In addition, since the depth map is estimated frame by frame, the temporal consistency cannot be guaranteed. This thesis proposes a method to enhance temporal consistency by applying adaptive temporal filtering, which explicitly considers the reliability of depth and the moving attribute of regions. Experiments demonstrate that the proposed algorithm can generate more stable depth sequences and effectively suppress the transient depth errors when rendering virtual images.

Due to the inherent drawbacks of stereo matching, the depth map captured by sensors is more robust, especially for the textureless regions. However, it either suffers from low resolution, or has some holes on the depth map. Active methods are to solve these problems. Since low-quality depth map is always captured with a high-quality color or intensity image and they can be registered with each other on the same coordinate system, low-quality

depth map can be refined by using the guidance from such high-quality color/intensity image. This type of active method is called guided depth map enhancement. In consideration of clear expression, this thesis uses color image to stand for color/intensity image in the rest of thesis. The meaning of it is according to the context.

The methods on guided depth map enhancement can be classified into different categories depending on whether external training data is used. Without relying on the external datasets, co-occurrence property between edges on the depth map and the corresponding color image is explicitly exploited. However, because the assumption above is not always true, it leads to texture-copy artifacts and blurring depth edges. Markov-Random-Field-based (MRF-based) methods are popular in guided depth map enhancement. The state-of-the-art solutions are to adjust the affinities of the regularization term in MRF energy function. Actually, these existing methods are lack of explicit evaluation model to quantitatively measure the inconsistency between the depth edges and the corresponding color edges, so they cannot adaptively control the efforts of the guidance from the color image for depth enhancement. In addition, widely used affinity computing scheme for regularization term is based on the depth and color differences between neighbor pixels, which ignores local structure on the depth map. In this thesis, three algorithms are proposed to address the problems above. The first one aims to mitigate artifacts caused by edge misalignment between the depth map and the color image via hard-decision inconsistency checking pixel by pixel. The second one uses a structural quantitative measurement on edges inconsistency which is a soft-decision method. It is more accurate than its hard-decision counterpart above. The third one is to combine such soft-decision edge inconsistency measurement and local structure of the depth map which is modeled on Minimum Spanning Trees (Forest) to acquire more robust depth map. These methods are tested on Middlebury, ToF-Mark and NYU datasets which prove progressive improvements.

In addition to the handcraft models for depth map enhancement, data-

driven models are expected to implicitly learning such guidance to obtain superior performances. In this thesis, an end-to-end training method based on convolutional neural network is proposed, which borrows many concepts from existing models, e.g., batch-normalization and residual learning. It upsamples low-resolution depth map progressively and the residual network is constructed to learn high frequency component in multiple scales. This coarse-to-fine scheme can reconstruct high-resolution depth via multi-frequency synthesis. Experimental results show improvement in subjective evaluation and objective evaluation compared with state-of-the-art methods.

# Chapter 1

# Introduction

## 1.1 Background

Depth represents geometric information of real scenes by recording the distances between the pixels and the camera, which can transform image pixels from a 2D coordinate system to 3D space (i.e., point cloud). Acquiring high-quality depth maps is the key problem in the field of 3D computer vision, which is required in many modern applications, e.g., interactive view interpolation, 3DTV, 3D object modeling, robot navigation, and 3D tracking. However, it is more difficult than conventional image acquisition. Generally speaking, methods for depth map acquisition consist of two categories: passive and active.

Passive methods can generate a depth map from two-view or multi-view color images using stereo matching algorithms. For decades, a variety of approaches have been proposed. Stereo matching methods can be categorized into two major classes: local and global. Local methods compute the depth value for each pixel independently through matching the local information only. Global methods determine the depth values of all the pixels simultaneously based on the predefined prior (e.g. the Total Variation norm). Local methods are faster than global ones but at the cost of accuracy. Over the last few decades, the performances of stereo matching are significantly im-


**(a)** occlusion


**(b)** textureless region

**Figure 1.1:** An illustration of occlusion and textureless regions

proved. However, these methods still suffer from the inherent problems such as matching difficulties in textureless areas and occlusion (Szeliski, Zabih, Scharstein, Veksler, Kolmogorov, Agarwala, Tappen & Rother 2008), which are shown in Fig. 1.1.

In real applications, it is important to balance running time and matching performance. In addition, since depth maps are estimated frame by frame, the depth values of the static background in adjacent frames cannot be consistent, which leads to artifacts (e.g. flashing artifacts when these depth maps are used to render virtual views of video.) This problem is called as temporal inconsistency (Fu, Zhao & Yu 2010).

Active depth-acquisition methods can obtain depth videos with the same frame rate as color cameras using depth sensors. Compared with passive methods, depth acquisition through active methods is much more efficient.


**(a)** structured-light sensor **(b)** ToF sensor

**Figure 1.2:** Some typical products for each type of sensor

Particularly, in textureless areas, active methods are able to achieve more robust performances than passive methods. So far, there are two main types of depth sensor, which are ToF sensors (e.g. SwissRanger 4000) and structured-light sensors (e.g. Kinect v1). Some typical products are shown in Fig. 1.2. By using ToF sensors, depth maps are computed by measuring the phase difference between the emitted light and the reflected light (Kolb, Barth, Koch & Larsen 2010). The drawback is that the captured depth maps are noisy with low resolutions, e.g., 176×144 or 200×200. By using structured-light sensors, an infrared light source projects a dot pattern on the scene. Another offset infrared camera senses the pattern and estimates the depth map. Although structured-light sensors can obtain depth maps with higher resolutions, the quality of the depth maps obtained by such sensors is not satisfying. There are holes (i.e., places without depth information sensed) appearing on the depth map. These holes may be caused by occlusion, weak reflection to the infrared light on surfaces or even shadow reflection of the light patterns. The depth maps captured by these types of sensors are illustrated in Fig. 1.3. Overall, objects in darker colors, specular surfaces, or fine-grained surfaces, e.g., human hair, are difficult to get depth sensing through depth sensors (Cho, Kim, Ho & Lee 2008). According to the analysis above, the main problems of depth maps obtained by depth sensors are low resolution, noisy depth values and holes. Such low-quality depth maps are always generated with a registered high-quality color or an intensity im-

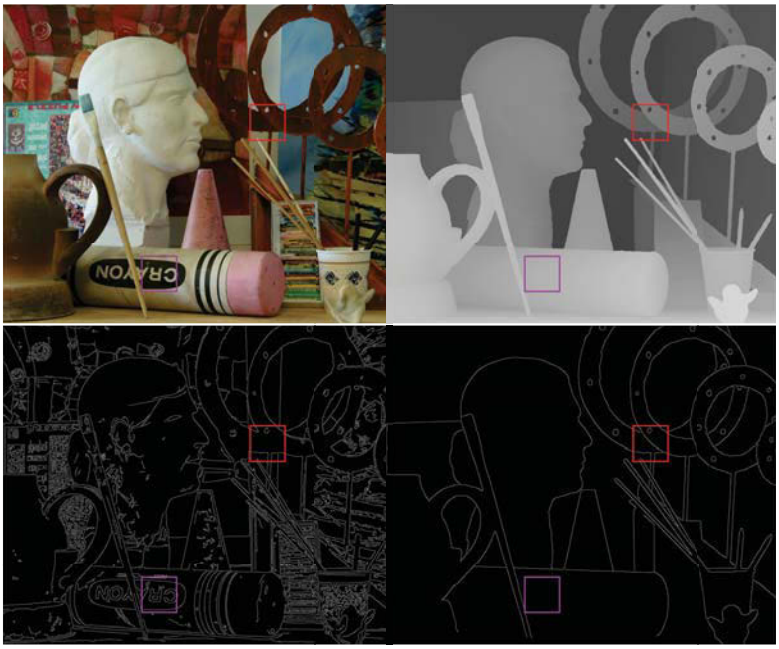
**(a)** depth captured by Kinect v1 **(b)** depth captured by ToF

**Figure 1.3:** An illustration of depth maps captured by sensors

age. Therefore, depth map can be enhanced guided by the corresponding color/intensity image due to high correlation between them. This type of method belongs to guided depth map enhancement. In consideration of clear expression, this thesis uses color image to stand for color/intensity image in the rest of thesis. The meaning of it is according to the context.

The existing methods for guided depth map enhancement can be classified into different categories depending on whether external training data is used. They correspond to solving predefined model via optimization or learning model from external data. Without relying on the external datasets, several approaches explicitly exploit co-occurrence property between the edges on the depth map and the corresponding color image (Diebel & Thrun 2005, Kopf, Cohen, Lischinski & Uyttendaele 2007). These methods show that the details and the accuracy of most depth edges can be improved based on the guidance information provided by the corresponding color edges. However, this assumption is not always true. Incorrect guidance from the companion color image will lead to texture-copy artifacts and blurring depth edges on the reconstructed depth map. Texture-copy artifacts derive from the situation that the smooth depth region corresponds to the color region with rich texture. By contrast, blurring depth edges result from the case that the smooth color region corresponds to the depth region with edges. Fig. 1.4 illustrates the edge inconsistency explained above.

For the works using external data, the ones based on sparse represen-



**(a)** color image and its edge map **(b)** depth map and its edge map

**Figure 1.4:** An illustration of edge inconsistency: pink window-edges occur on the color image but not on the depth map, red window-edges occur on the depth map but not on the color image.

tation are firstly proposed (Li, Xue, Sun & Liu 2012, Kiechle, Hawe & Kleinsteuber 2013, Kwon, Tai & Lin 2015). In such approaches, the corresponding image patches acquired from the low-resolution (LR) depth map, high-resolution (HR) depth map and color image can be represented using a sparse vector on corresponding learned generic dictionaries respectively. Under the assumption that the sparse vector is shared among these registered patches, statistical dependencies between color and depth patches can be learned by joint sparse coding scheme. Very recently, a few works are based on end-to-end Convolutional Neural Networks (CNN) (Riegler, Ferstl, Rüther & Bischof 2016, Hui, Loy & Tang 2016). Compared with model-driven methods which explicitly exploit co-occurrence property between edges on depth map and corresponding color image, such data-driven methods enhance low-

quality depth map by using guidance from registered color image implicitly. Due to the different statistical attributes, texture-copy artifacts and depth blurring edges may also occur in such machine learning methods. They largely rely on the similarity between training data distribution and testing data distribution.

## 1.2 Research Problems

Based on the aforementioned current research limitations, this thesis presents the following research problems.

### 1.2.1 Fast Depth Map Estimation

Compared with global methods, local methods (Olgierd Stankiewicz 2009, Zhang, Lu & Lafruit 2009, Hosni, Bleyer, Rhemann, Gelautz & Rother 2011) have low complexity. Therefore, it is meaningful to improve the performance of local methods, especially for the real-time applications. Among such methods, adaptive-weight-based methods always reach the best results. The weight represents how likely the current pixel sharing the similar disparity with the central pixel of the local window. Most adaptive-weight-based methods assume that such weight is according to the color difference and the pixel location distance between the current pixel and the central pixel. Based on such weights, the effort of each neighbor pixel in the matching cost aggregation can be adaptively controlled.

Despite the excellent results produced by adaptive-weight-based methods, pixel-wise weight computation and depth estimation are highly time-consuming tasks. They have computational redundancy because depth maps are always smooth. If smoothness attribute could be used to replace pixel-wise depth estimation, the running time would be significantly reduced. The related analysis and solution are explained in Chapter 3, which proposes a robust local method reducing the running time.

### 1.2.2 Temporal Consistency Enhancement for Depth Video

Depth sequences generated by single-frame depth estimation suffer from the temporal inconsistency problem. Ideally, the depth values of static objects should remain the same in adjacent frames, but they are often estimated as being different values. These temporal depth errors significantly degrade the visual quality of the synthesized virtual view as well as the coding efficiency of the depth sequences (Fu et al. 2010, Sang-Beom Lee & Ho n.d.). In this thesis, a corresponding temporal consistency enhancement method is proposed in Chapter 3.

### 1.2.3 Explicit Measurement for Edge Inconsistency between Depth Map and Color Image

Among the guided depth-enhancement methods without using the external datasets, there have been several ones (Park, Kim, Tai, Brown & Kweon 2014, Yang, Ye, Li, Hou & Wang 2014, Choi & Jung 2014, Hua, Lo & Wang 2016) trying to mitigate texture-copy artifacts and blurring depth edges via balancing the contribution from the original depth map and the companion color image. However, existing methods do not explicitly evaluate the edge inconsistency between the color image and the corresponding depth map. Therefore, they cannot adaptively control the guidance efforts from the color image when enhancing the depth map. Some texture-copy artifacts and blurring depth edges still appear on their results. A quantitative measurement which provides a more precise definition of the inconsistency in a numerical way is needed. In this thesis, Chapter 4 shows two edge inconsistency measurement models via hard-decision and soft-decision manners.

### 1.2.4 Improved Guided Depth Enhancement via Predefined Model

If the explicit edge inconsistency measurement model above can be embedded into a predefined model, e.g. Markov Random Field (MRF), the texture-copy artifacts can be significantly mitigated. In addition, the edge guidance affinities of MRF regularization term are usually computed only based on color and depth differences between the pixel and its neighbor pixels on color image and coarsely interpolated depth map respectively in existing methods. Such computing scheme ignores the local structure of the depth map and is called as non-structural scheme throughout this thesis. Therefore, in the case of a large upsampling factor, it generates over-smoothed depth edges. A more robust model is needed to preserve structure in extremely low-quality depth maps. Such problems are solved in Chapter 4.

### 1.2.5 Guided Depth Enhancement via Machine Learning

Predefined models use prior regularization that comes from general cases to solve ill-posed problems. However, for certain types of signal, such predefined models may not provide optimal performance. Instead, the model learning from the data itself may lead to more robust results (Hui et al. 2016, Riegler et al. 2016). Since the guidance from the color image is implicitly used, it is important to learn the inherent different statistical characteristics between the depth map and the registered color image in the trained model. Such attempt can improve the performance of depth map super-resolution. The corresponding experiments are shown in Chapter 5

## 1.3 Research Contributions

- Proposed a local algorithm to fast estimate depth maps by combining an adaptive matching scheme with an affine invariant feature. It can

achieve better or comparable performances than the state-of-the-art approaches in the category of local methods, even with less running time. (Chapter 3)

- Proposed an algorithm to enhance temporal consistency for estimated multi-view depth videos. It effectively suppresses the transient depth errors and generates more stable depth sequences via an anisotropic temporal filter. (Chapter 3)

- Proposed a pixel-level hard-decision and soft-decision structure-measurement models to evaluate edge inconsistency between depth edges and the corresponding color edges. The soft-decision model can provide a more precise definition of the edge inconsistency in a numerical way than the hard-decision counterpart. (Chapter 4 )

- A novel guidance affinity computing scheme for the regularization term in MRF is proposed to better preserve edges and structure. The Affinities are computed more precisely in a space which consists of multiple Minimum Spanning Trees (MSTs). Soft edge inconsistency measurement is considered and embedded into edge weights in each MST to significantly mitigates texture-copying artifacts. (Chapter 4)

- Proposed a modified end-to-end convolutional residual network leaning the guidance from the registered color image to progressively upsample low resolution noisy depth map. (Chapter 5)

## 1.4 Thesis Structure

The thesis is structured as follow:

Chapter 2 firstly introduces the basic knowledge for depth estimation and depth enhancement. Then, local optimization theory, e.g., typical filters, as well as the global optimization theory, e.g., Markov Random Field (MRF), are introduced with the representative works for such tasks. In addition, this

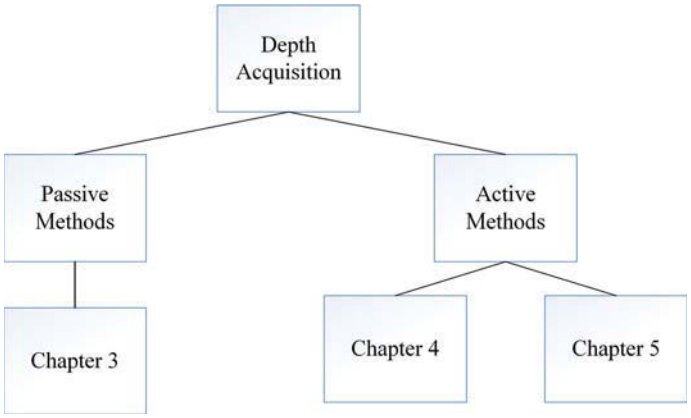



**Figure 1.5:** The relation among Chapter 3, 4 and 5

chapter reviews the related theory of machine learning (i.e., sparse coding and convolutional neural network) followed by the representative works for guided depth map super-resolution.

Chapter 3 firstly briefly reviews the related work for depth estimation and explains the motivation of the proposed method. In the following section, an adaptive matching scheme for multi-view depth video estimation is proposed. Then, the affine invariant feature is proposed to make the matching more robust for low-texture regions. At last of this chapter, a temporal enhancement algorithm is proposed for multi-view depth video estimated frame by frame.

Chapter 4 briefly reviews existing methods for depth map enhancement and provides the motivation of this chapter. Subsequently, hard-decision and soft-decision edge inconsistency measurement models are proposed respectively. Such models are embedded into the non-structural regularization term in MRF energy function. To further improve performance, a structural affinity computing scheme is proposed to replace the non-structural one.

Chapter 5 begins with briefly reviewing learning-based methods for depth map super-resolution. In the following, it presents a modified convolutional neural network learning the guidance from the HR color image to progressively upsample the LR noisy depth map.

The relation among Chapter 3, 4 and 5 is illustrated as Fig. 1.5. In

addition, the thesis adopts popular Peak signal-to-noise ratio (PSNR) and root-mean-square error (RMSE) which are equivalent as the metric in Chapter 3 and 5 respectively. Both of them are based on L2 norm. However, due to the blurry results are preferred under L2 norm, the results are evaluated by Mean Absolute Error (MAE) which is based on L1 norm.

Chapter 6 concludes the thesis and provides a discussion for future work.

# Chapter 2

# Relevant Theories and Related Work

The chapter firstly reviews the foundation in passive depth acquisition and active depth acquisition in Section 2.1 and Section 2.2 respectively. Section 2.3 presents classical filters and their usages in such tasks. Markov Random Field (MRF) and its configurations for such applications are reviewed in Section 2.4. Section 2.5 presents related machine learning theory for depth enhancement only since this thesis does not focus on learning-based depth estimation method. Finally, Section 2.6 ends this chapter with a summary.

## 2.1 Foundation for Passive Depth Acquisition via Stereo Matching

By given multi-view color images of the same scene, each pixel in certain viewpoint and their corresponding pixels in other viewpoints can be determined by stereo matching. The displacement vector between each registered pixel pair is called as disparity. This section introduces concepts, constraints, basic procedure and refinement for depth estimation.

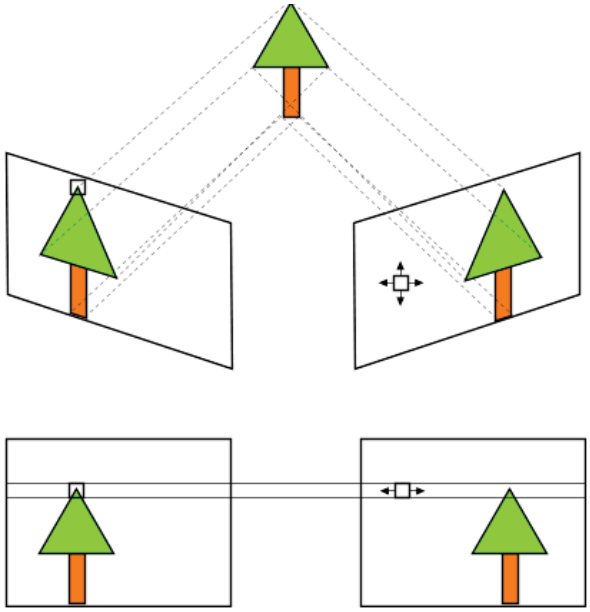


**Figure 2.1:** An illustration of image rectification

### 2.1.1 Concepts

Related concepts include image rectification and the relation between disparity and depth. Image rectification is the necessary pre-processing step to reduce the dimension of disparity from two to one. In addition, since the output of stereo matching is disparity, the relation between it and depth should be addressed.

#### Image Rectification

The purpose of image rectification is to make the epipolar lines of two images horizontally aligned. It can be accomplished through linear transformations of images using internal and external camera parameters. After image rectification, the matching candidates locate in the same row as the pixel to be matched. Therefore, the searching dimension is reduced from two to one. Fig. 2.1 gives an illustration of image rectification. In the rest of this chapter, it is assumed that image pairs are rectified beforehand.

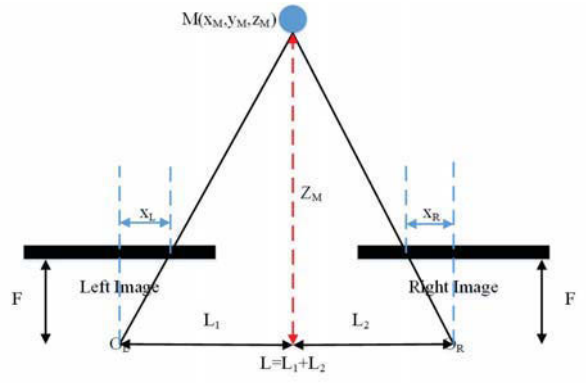



**Figure 2.2:** The relation between disparity and depth

### Relation between Disparity and Depth

The relation between disparity and depth for rectified image pair is shown in Fig. 2.2. From this figure, disparity $d$ of point $M$ can be computed as Eq. (2.1);

$$d = x_L - x_R = \frac{FL_1}{Z_M} + \frac{FL_2}{Z_M} = \frac{FL}{Z_M} \tag{2.1}$$

where $F$ and $L$ represent focal length and baseline length respectively. $x_L$ and $x_R$ are $x$ coordinates of projections on two planes. $Z_M$ is $z$ coordinate of point $M$. This equation means that once the disparity is computed, corresponding depth can be found. In the rest of this thesis, disparity and depth is equivalent.

## 2.1.2 Stereo Matching Constraints

Since dense depth estimation based on stereo matching is ill-posed, some prior should be introduced to reduce ambiguity. This part presents some widely used priors.

#### Intensity Similarity

One of the basic priors is intensity similarity which means corresponding pixels belonging to the same part of an object in multiple images have similar intensity value. This prior will fail when the illumination conditions for cameras are different.

#### Uniqueness

Uniqueness prior means all the pixel matching is one-one matching. This prior is not always true, e.g. occlusion.

#### Smoothness

Disparity map (depth map) is smooth along the surface of objects. The discontinuities only occur at the edge of objects.

### 2.1.3 Basic Procedure

Basic procedure of depth estimation includes cost aggregation and disparity computing. They are briefly introduced as below.

Typically, stereo matching is performed within a certain disparity searching range. The quality of matching which maps a pixel in current view to the pixel in other view determined by disparity is evaluated by a matching cost defined by certain cost function. The cost volume consists of matching costs of all pixels under all disparity candidates in the searching range. This procedure is called cost aggregation. After computing such cost volume, the disparities can be determined by local or global methods. In the following parts, such components are explained in detail.

#### Cost Aggregation

For a certain disparity $d$ in the predefined search range $[d_{min}, d_{max}]$, there is a matching cost between the reference pixel $p$ with coordinate $(p_x, p_y)$ and its matching candidate $q$ with coordinate $(q_x = p_x + d, q_y = p_y)$. For simplicity,

$p$ and $q = p + d$ state for matching pixel pair above in the rest of thesis. Since single pixel matching is sensitive to noise, the cost is computed between two local windows which are centered by $p$ and $q$. Many cost functions are proposed corresponding to different evaluation measurement. Based on intensity similarity prior, sum of absolute intensity differences (SAD), sum of squared intensity differences (SSD) are proposed. According to features in gradient field, sum of absolute gradient differences (SGRAD) is presented to complement SAD or SSD when the intensity similarity prior fails. These cost functions are shown in Eq. (2.2), Eq. (2.3) and Eq. (2.4).

$$Cost_{SSD}(p,d) = \sum_{s \in \mathbf{N_p}} |\mathbf{I_L}(s) - \mathbf{I_R}(s+d)| \tag{2.2}$$

$$Cost_{SAD}(p,d) = \sum_{s \in \mathbf{N_p}} (\mathbf{I_L}(s) - \mathbf{I_R}(s+d))^2 \tag{2.3}$$

$$\begin{aligned} Cost_{SGRAD}(p,d) = \sum_{s \in \mathbf{N_p}} & |\nabla_x \mathbf{I_L}(s) - \nabla_x \mathbf{I_R}(s+d)| + \\ & |\nabla_y \mathbf{I_L}(s) - \nabla_y \mathbf{I_R}(s+d)| \end{aligned} \tag{2.4}$$

where $\mathbf{I_L}$ and $\mathbf{I_R}$ are images in different viewpoints. $\mathbf{N_p}$ is a local window centered at $p$. $\nabla_x$ and $\nabla_y$ are gradient operators. $d$ is an element in disparity searching range $[d_{min}, d_{max}]$. More advanced cost functions are shown in next chapter.

### Disparity Computing

Based on computed the cost volume $Cost(p,d)$ for each pixel $p$ and each disparity $d$, the optimal disparities for all pixels can be determined via local or global methods. The local methods determine disparity $d_p$ for each pixel $p$ independently to find the best matching which is modeled as Eq. (2.5);

$$d_p = \arg\min_d Cost(p,d) \tag{2.5}$$

Compared with local methods, the global methods always explicitly place certain regularization on the disparity map, e.g., smoothness prior. And the optimal disparities $\mathbf{D}^\star$ for all pixels are determined simultaneously via global energy function optimization. This scheme has the form as Eq. (2.6),Eq. (2.7) and Eq. (2.8).

$$\mathbf{D}^\star = \arg\min_{\mathbf{D}} data\left(\mathbf{D}\right) + \lambda reg\left(\mathbf{D}\right) \tag{2.6}$$

$$data\left(\mathbf{D}\right) = \sum_{d_p \in \mathbf{D}} Cost\left(p, d_p\right) \tag{2.7}$$

$$reg\left(\mathbf{D}\right) = \sum_{i} \sum_{j \in \mathbf{N_i}} V\left(d_i, d_j\right) \tag{2.8}$$

where $\mathbf{N_i}$ is the neighborhood of pixel $i$, $data$ and $reg$ are data term and regularization term respectively. $\lambda$ is the balance factor. A common idea of $reg$ is to let $V$ increase monotonically with disparity difference to penalize a discontinuous result, while be able to reduce this penalty located at color edges to preserve depth edges. Such methods are related to graphic models e.g., Markov Random Field (MRF). More information on MRF are shown in Section 2.4.

### 2.1.4 Refinement

Since depth varies continuously almost everywhere and that depth discontinuities occur primarily at color edges, depth values can be approximately modeled as a plane within segments. This refinement technique is called plane fitting. Another widely used post-precessing is cross-check which validates disparities by using disparity maps in multiple viewpoints. This part shows such refinement techniques.

#### Plane Fitting via RANSAC

The depth map processed by plane fitting always presents smoothness attribute. However, such fitting maybe difficult due to the noise and outliers

in the initial estimated depth map. Compared with least squares, techniques such as Random Sample Consensus (RANSAC) (Choi, Kim & Yu 1997) is more robust to outliers.

RANSAC alternates between two steps, hypothesizing and testing, with a model definition and an evaluation function which provides distance between a data point and the model. The hypothesis step is to fit the model by using the minimum amount of sample points required to fully define the model parameters which are randomly selected. The testing step measures the distance between all other data points and the model via evaluation function. A threshold is defined to classify inliers and outliers. The model has most inliers is assumed to be the most representative one. As an optional step, the model can be refitted using all the found inliers.

In the case of plane fitting, the used model and evaluation function $E$ are the plane equation and the distance between data point $\mathbf{P}$ with coordinate $(p_x, p_y, p_z, 1)$ to plane $\mathbf{T}$ represented by normal vector $(t_x, t_y, t_z, 1)$ which are shown as Eq. (2.9) and Eq. (2.10) respectively.

$$\mathbf{S} \cdot \mathbf{T} = 0 \tag{2.9}$$

where $\mathbf{S}(s_x, s_y, s_z, 1)$ represents arbitrary point on the plane $\mathbf{T}$.

$$E = \frac{\mathbf{P} \cdot \mathbf{T}}{\sqrt{t_x^2 + t_y^2 + t_z^2}} \tag{2.10}$$

The number of iterations is a self-adapting parameter which depends on the probability $pg$ of finding a good sample set and the maximum allowed probability $pb$ to only pick bad sample sets. The number of iterations $iter$ is determined by $(1 - pg)^T \leqslant pb$. For more information, please refer to (Choi et al. 1997).

### Cross check between Two Disparity Maps

Disparity map represents the displacements from the pixels in the reference image to the corresponding pixels in the target image. Ideally, the disparity map which swap the reference image and the target image is equivalent to

the original one. However, unreliable estimated disparities for certain set of pixels on such disparity maps are always conflict with each other. Therefore, disparity maps for multiple viewpoints can be used to check the reliability of disparity for each pixel. More specifically, the reliability of them is determined by comparing disparities of the reference pixels against the ones of target pixels on other views which is defined as Eq. (2.11).

$$\begin{aligned} \mathbf{D_L}(x,y) &= \mathbf{D_R}(x - \mathbf{D_L}(x,y), y) \\ \mathbf{D_R}(x,y) &= \mathbf{D_L}(x + \mathbf{D_R}(x,y), y) \end{aligned} \tag{2.11}$$

where $\mathbf{D_L}$ and $\mathbf{D_R}$ are the disparity maps for the left image and the right image respectively. $x, y$ are coordinate in the disparity maps. If certain pixels do not satisfy this checking, they are unreliable which can be refined by the filters shown in Section 2.3. The underlying idea for this refinement is to correct unreliable values through nearby reliable values.

## 2.2 Foundation for Active Depth Acquisition via Sensors

Alternatively, depth maps can be acquired by sensors, which is more robust to textureless regions compared with passive depth acquisition. However, the quality of such depth maps do not satisfy applications. So, the enhancement is necessary for this type of acquisition. In this thesis, depth enhancement includes depth map super-resolution (SR) and depth map completion which are corresponding to the depth maps obtained by ToF sensors and structural-lighted sensors. Firstly, the common degradation model is introduced for the two problems above. Subsequently, the related works are briefly reviewed.

### 2.2.1 Degradation Model

Suppose $\mathbf{Y}$ is degraded signal or image, its degradation model can be defined as Eq. (2.12).

$$\mathbf{Y} = \mathbf{HX} + \mathbf{N} \tag{2.12}$$

where **H** represents degradation matrix corresponding to downsampling and hole generation. **X** is original high-quality signal. **N** stands for additive noise.

Such model gives the relation between high-quality depth map and its low-quality version, which is used in depth reconstruction. This model shows that depth enhancement is an ill-posed inverse problem.

### 2.2.2 Related Work

Since depth map SR and depth map completion can be formulated as the same inverse problem above, this thesis reviews common methods for such two tasks. Existing methods can be classified into two categories: non-guided methods (Freedman & Fattal 2011, Schuon, Theobalt, Davis & Thrun 2009, Xie, Feris & Sun 2016) and guided methods (Hua et al. 2016, Yang, Ye, Li, Hou & Wang 2014, Park et al. 2014).

#### Non-guided Methods

Since the thesis focuses on guided depth enhancement, non-guided methods are briefly introduced as follows; (Freedman & Fattal 2011) only requires a single depth map for SR by using smoothing priors based on local self-similarities, but it either has difficulties in textured areas, or only works well for the case of small upsampling factor. (Xie et al. 2016) proposed a single depth map SR method via a modified joint bilateral filter. Such bilateral filter is guided by a high-resolution (HR) edge map which is constructed from the edges of the low-resolution (LR) depth map through an MRF optimization in a patch synthesis based manner. Another type of non-guided approach (Schuon et al. 2009) is to fuse multiple displaced LR depth maps into a single HR depth map, which is not convenient for real applications because the geometrical relationship between all the depth sensors cannot be easily determined.

### Guided Methods

Guided methods intend to improve the quality of the captured depth map with the support of a registered color image. Such guided methods can be classified into three categories that are local methods (Kopf et al. 2007, Liu, Tuzel & Taguchi 2013, He, Sun & Tang 2010, Min, Lu & Do 2012), global methods (Diebel & Thrun 2005, Ferstl, Reinbacher, Ranftl, Rüther & Bischof 2013, Yang, Ye, Li, Hou & Wang 2014, Park et al. 2014) and learning-based methods (Li et al. 2012, Kiechle et al. 2013, Riegler et al. 2016, Hui et al. 2016).

Local methods typically refine depth for each pixel independently which interpolates current unavailable pixels through weighted average. The adaptive weight is guided by registered color image. Global methods perform enhancement through modeling this problem into an optimization of the predefined energy function. They always based on graphic models which places certain prior guided by color image on the depth map to reduce the ambiguity of such ill-posed inverse problem. The fundamental assumption of local and global methods is that the depth edges and the color edges at the corresponding locations are consistent. By using external datasets, the guidance and prior attribute for specific type of signal can be learned. It is the directly expanding from single image enhancement.

Since there are common techniques which are related to both depth estimation and depth enhancement, they are shown in details in the following parts. These techniques consist of filters, graphic model, global optimization and related machine learning theories.

## 2.3 Local Methods and Filters

After introducing the basic concepts, this section presents classical filters and their usages in depth estimation and depth enhancement.

### 2.3.1 L2 Norm Optimization Filters

For representative filters, bilateral filtering techniques (Tomasi & Manduchi 1998) should be presented. It has form of Eq. (2.13);

$$e'_p = \sum_{q \in \mathbf{N_p}} \mathbf{W}(q,p) \times e_q \tag{2.13}$$

where $e_q$ is the element used for weighted average which has different meanings for depth estimation and depth enhancement. More details are introduced in the following paragraphs. $\mathbf{W}$ is the weighting kernel which is defined as Eq. (2.14). $\mathbf{N_p}$ represents the local window which is centered at $p$. $e'_p$ states for the filtered value.

$$\mathbf{W}(q,p) = e^{-\frac{(\mathbf{I}(p)-\mathbf{I}(q))^2}{2\delta_c^2}} \times e^{-\frac{(p-q)^2}{2\delta_d^2}} \tag{2.14}$$

where $\delta_c, \delta_d$ controls bandwidth of range and distance kernels respectively, $\mathbf{I}$ is the color image and $\mathbf{I}(p), \mathbf{I}(q)$ are its elements. $p, q$ represent the 2-dimension pixel locations on the image. Essentially, this filter is an anisotropic weighted average. The weight for each pixel pair is computed by color difference and Euclidean distance, which correspond to two parts in Eq. (2.14). More specifically, it based on a reasonable assumption that the elements corresponding to pixels which have similar color and location with center pixel have heavy weight in average.

For edge-preserving smoothing, $e_q$ is the value for pixel $q$ in arbitrary channel (Bao, Song, Yang, Yuan & Wang 2014). It can prevent pixels which are very different in color with central pixel participating in average. Therefore, edges can be preserved while smoothing the image.

For cost aggregation in stereo matching, $e_q$ is the pixel matching cost with certain disparity $d$ in a local window (Yang, Wang, Yang, Stewénius & Nistér 2009). The final matching cost is weighted average of those pixel matching cost. It is assumed that pixels which are similar in color with central pixel have similar disparity values and they should play dominating role in weighted average. This scheme is also called soft segmenta-

tion (Olgierd Stankiewicz 2009) which is shown as Eq. (2.15).

$$Cost\left(p,d\right)=\frac{\sum\limits_{q\in\mathbf{Np}}\mathbf{W_{ref}}\left(q,p\right)\mathbf{W_{tar}}\left(q-d,p-d\right)diff\left(q,q-d\right)}{\sum\limits_{q\in\mathbf{Np}}\mathbf{W_{ref}}\left(q,p\right)\mathbf{W_{tar}}\left(q-d,p-d\right)} \tag{2.15}$$

where $\mathbf{W_{ref}}$ and $\mathbf{W_{tar}}$ are weights computed by bilateral filter (Eq. (2.14)) for reference window and target window respectively. $diff$ is pixel-level matching cost. Denominator is normalization factor.

For depth map SR, a joint bilateral upsampling (JBU) framework is proposed by Kopf et al. (Kopf et al. 2007), The edges of the LR depth map can be refined according to the edges of the registered HR color image. $e_q$ is the unfiltered depth value for pixel $q$. $e'_p$ states for the filtered value for pixel $p$.

Based on this pioneer work, some variants are proposed. Liu et al. (Liu et al. 2013) propose filter which computes weighting coefficients based on geodesic. Geodesic is computed in a joint space of color and distance instead of separating color space and distance space. Compared with JBU, joint geodesic kernel integrates color changes along the geodesic curves; therefore, it is more sensitive to thin structures and fine scale changes, producing smooth surfaces with sharp occlusion boundaries. He et al. (He et al. 2010) propose a guided image filtering for depth enhancement, which models a linear relationship between the output and guiding image. It is based on the assumption that the output has an edge only if the input has an edge. To determine the linear coefficients, it seeks a solution that minimizes an energy function. It is formed from the total square difference between the filtered image and the input image with a regularization parameter to prevent too large coefficients. All these methods including JBU have the same form with different weighting kernels as Eq. (2.13). The kernels of Liu et al. (Liu et al. 2013) and He et al. (He et al. 2010) are defined as Eq. (2.16) and Eq. (2.17) respectively. The kernel of Guided filter also can be used for cost aggregation in depth estimation (Yang, Ji, Li, Yao & Zhang 2014).

$$\mathbf{W}(q,p)=e^{-\frac{G_d(q,p)}{2\delta^2}} \tag{2.16}$$

where $G_d(q,p)$ is geodesic distance between $p,q$.

$$\mathbf{W}(q,p) = \frac{1}{|\omega|^2} \sum_{k:(p,q)\in\mathbf{N_k}} \left(1 + \frac{(\mathbf{I}(p) - \mu_k)(\mathbf{I}(q) - \mu_k)}{\sigma_k^2 + \epsilon}\right) \tag{2.17}$$

where $\mu_k$ is the mean of the local window centered by $k$. $k : (p,q) \in \mathbf{N_k}$ states for all the windows which include $p,q$. $|\omega|$ is the number of pixels in the local window $\mathbf{N_p}$. $\epsilon$ is a small value to prevent denominator from zero.

### 2.3.2 L1 Norm Optimization Filters

These kernels introduced above are based on L2-norm optimization. Alternatively, motivated by cost aggregation in stereo matching, some approaches based on L1 norm minimization is proposed which are more robust to outliers than L2 norm minimization. Yang et al. (Yang, Ahuja, Yang, Tan, Davis, Culbertson, Apostolopoulos & Wang 2013) propose a depth map SR method based on joint bilateral filtering (JBF) techniques with a set of depth candidates for iteratively refining HR depth map. The final depth value is selected by using the winner-takes-all (WTA) method on the cost volume after a predefined number of iterations. The experiment shows that it can give a better edge-preserving performance. Min et al. (Min et al. 2012) propose a weighted mode filtering method based on a joint histogram. When the histogram is generated, the weight based on color similarity between reference pixel and its neighboring ones on the color image is computed. This weight is used for counting each bin on the joint histogram. A final solution is determined by seeking a global mode on the histogram. They share the form of Eq. (2.18) and Eq. (2.19). From these equations, the similarity between such type of filter and cost aggregation in stereo matching Eq. (2.5) is shown. Again, they share the same assumption that pixels which are similar in color have similar depth value.

$$d_p^\star = \underset{d\in[d_{min},d_{max}]}{\arg\max}\ Cost(p,d) \tag{2.18}$$

$$Cost(p,d) = \sum_{q\in\mathbf{N_p}} \mathbf{W}(q,p)\, Er(d - d_q) \tag{2.19}$$

where *Cost* is the cost volume. Under certain label $d$, the cost value for each pixel $p$ is computed by weighted mean. $\mathbf{W}(q,p)$ is the same meaning as Eq. (2.14). $Er$ is the error function which is the difference between methods of this type. Yang et al. (Yang et al. 2013) and Min et al. (Min et al. 2012) construct the error function as Eq. (2.20) and Eq. (2.21) respectively.

$$Er\left(d-d_q\right)=\min\left(\left|\left(d-d_q\right)\right|,\tau\right) \tag{2.20}$$

$$Er\left(d-d_q\right)=e^{-\frac{\left(d-d_q\right)^2}{2\delta^2}} \tag{2.21}$$

where $\tau$ and $\delta$ are predefined parameters.

All the local methods can be performed iteratively which is a coarse-to-fine manner. Overall, the complexity of local methods is low, so the efficiency is attractive. However, they are always inferior to global counterparts especially for the noise-aware case.

## 2.4 Global Optimization and Graphic Models

### 2.4.1 Relation to Bayesian inference

Compared with local methods introduced in Section 2.3, global counterparts are more robust. The specific task is modeled as an optimization problem of the predefined objective function. Such function is also called as energy function. Almost global methods belong to Bayesian inference. For example, the general model for Markov Random Field (MRF) is defined as Eq. (2.22) according to the Hammersely-Clifford theorem (Hammersley & Clifford 1971).

$$\begin{aligned} pb\left(\mathbf{X}|\mathbf{Y}\right)&=\frac{pb\left(\mathbf{Y}|\mathbf{X}\right)pb\left(\mathbf{X}\right)}{\sum\limits_{\mathbf{X}}pb\left(\mathbf{Y}|\mathbf{X}\right)pb\left(\mathbf{X}\right)}\\ pb\left(\mathbf{Y}|\mathbf{X}\right)&=e^{-data(\mathbf{X},\mathbf{Y})}\\ pb\left(\mathbf{X}\right)&=e^{-\lambda reg(\mathbf{X})} \end{aligned} \tag{2.22}$$

where $\mathbf{X}$ is the inference result (i.e., high-quality depth map for depth enhancement or depth estimation). $\mathbf{Y}$ is an observed information for inference (e.g., observed depth values for depth enhancement, matching error for depth estimation). *pb* is abbreviation for unnormalized probability. $data\left(\mathbf{X},\mathbf{Y}\right)$ represents data term in energy function which describes appearing probability of certain status of $\mathbf{Y}$ given $\mathbf{X}$. $reg\left(\mathbf{X}\right)$ is regularization term which models prior distribution of $\mathbf{X}$ itself. $\lambda$ is balance factor between them.

Based on the theory of maximum a posteriori estimation, the inference can be solved by $\arg\max_{\mathbf{X}} pb\left(\mathbf{X}|\mathbf{Y}\right)$. Since the denominator is only related to $\mathbf{Y}$, it can be ignored. So the equivalent solution is to maximize $\hat{pb}\left(\mathbf{X}|\mathbf{Y}\right)$ defined as Eq. (2.23);

$$\mathbf{X}^{\star}=\arg\max_{\mathbf{X}}\hat{pb}\left(\mathbf{X}|\mathbf{Y}\right)=\arg\max_{\mathbf{X}}pb\left(\mathbf{Y}|\mathbf{X}\right)pb\left(\mathbf{X}\right) \tag{2.23}$$

Therefore, minimizing the energy function $E$ which is defined as Eq. (2.24) is equivalent to maximize $\hat{pb}\left(\mathbf{X}|\mathbf{Y}\right)$.

$$\mathbf{X}^{\star}=\arg\min_{\mathbf{X}}E\left(\mathbf{X},\mathbf{Y}\right)=data\left(\mathbf{X},\mathbf{Y}\right)+\lambda reg\left(\mathbf{X}\right) \tag{2.24}$$

The regularization term $reg$ corresponds to certain prior. For example, according to smoothness prior, $reg$ can be modeled as Eq. (2.25).

$$pb\left(\mathbf{X}\right)=e^{-\lambda reg\left(\mathbf{X}\right)}=e^{-\left\|\mathbf{Vec}\left(\nabla\mathbf{X}\right)\right\|_{2}} \tag{2.25}$$

where $\nabla$ is gradient operator which computes differences between each pixel and their nearby pixels. **Vec** is vectorized operator which reshapes all gradients in a vector. The typical definitions of data term *data* is illustrated in the next subsection for specific tasks.

### 2.4.2 Bayesian Inference for Depth Estimation and Depth Enhancement

For depth estimation, $\mathbf{X}$ and $\mathbf{Y}$ are inferred depth $\mathbf{D}$ and observed matching cost $\mathbf{Cost}\left(\mathbf{D}\right)$ for all pixels respectively. More specifically, for certain pixel $p$,

the true depth $d_p \in \mathbf{D}$ is more likely to correspond to lowest cost $Cost\,(d_p)$. Therefore, $pb\,(\mathbf{Cost}\,(\mathbf{D})|\mathbf{D})$ can be modeled as Eq. (2.26). It means that higher the cost is, lower the probability of true depth selected.

$$pb\,(\mathbf{Cost}\,(\mathbf{D})|\mathbf{D}) = e^{-data(\mathbf{D})} = e^{-\mathbf{Cost}(\mathbf{D})} \tag{2.26}$$

For depth enhancement, $\mathbf{X}$ and $\mathbf{Y}$ are inferred depth $\mathbf{D}$ and observed depth values $\mathbf{O}$ respectively. And $pb\,(\mathbf{O}|\mathbf{D})$ is related to the difference between $\mathbf{D}$ and the observed depth $\mathbf{O}$ which is shown as Eq. (2.27). The evaluation function $f$ can be specifically defined (e.g. MSE). It is said that the inferred depth values should be consistent with observed ones.

$$pb\,(\mathbf{O}|\mathbf{D}) = e^{-data(\mathbf{D},\mathbf{O})} = e^{-f(\mathbf{D},\mathbf{O})} \tag{2.27}$$

In the following parts, more details for depth estimation and depth enhancement via global optimization are introduced.

### 2.4.3 Discrete Optimization via Graph Cut

Because depth estimation is to estimate discrete disparity variables, it is a discrete optimization problem. After quantizing real depth value, depth enhancement can be discrete optimization problem as well. Graph cut (Boykov, Veksler & Zabih 2001) is a popular discrete energy minimization algorithm based on graph which are explained in this subsection. To simplify explanation modeling depth estimation and depth enhancement as a graph optimization problem, the basic concept of graph cut is introduced firstly.

The first concept is that graph cut solves binary discrete optimization problem via relabeling all variables. There are two relabel strategies which are $\alpha, \beta - swap$ and $\alpha - expansion$. $\alpha, \beta$ stand for possible binary values for variables. $\alpha, \beta - swap$ optimally relabels pixels via changing labels between $\alpha$ and $\beta$. $\alpha - expansion$ simply expands the set of pixels with label $\alpha$ to minimize energy function. Since the $\alpha - expansion$ is strongly favored in the survey (Szeliski et al. 2008), hereafter, only widely used $\alpha - expansion$ is introduced. For more detail on the differences between them, please refer

to (Boykov et al. 2001). Such operation is carried out continuously until the energy is convergent. The initial values of labels can be predefined or random. In this part, single low-case letters represent pixels e.g. $p, q$. The labels for them are defined as $d_p, d_q$.

The second concept is that the relabeling is performed by cutting edges linking nodes and two terminal nodes $\alpha, \alpha'$. More specifically, when the edge linking node and $\alpha$ is cut, that node's label is relabeled to $\alpha$. The label of the node is unchanged when the edge linking node and $\alpha'$ is cut.

The third concept is that a valid cutting plan is to separate two terminal nodes $\alpha, \alpha'$. It means that there is no path between them after cuts. For each cutting plan, the cost is the sum of weights of all the cutting edges. Among these solutions, the one with lowest cost is preferred.

For depth estimation, the labels are disparity candidates in searching range. And labels for depth enhancement are quantization levels. Therefore, such tasks are multi-label optimization problems and cannot be optimized via graph cut directly. Alternatively, the multi-label problem is decomposed into many binary discrete optimization sub-problems. Then, the original problem is approximately solved via solving sub-problems in order. In this subsection, depth estimation solved by graph cut is explained in detail.

Now the detail of modeling solving energy function Eq. (2.24) for depth estimation as graph cut is illustrated. The graph consists of a set of nodes, each representing a pixel. Two terminal nodes $\alpha, \alpha'$ are connected to all nodes via edges. There are edges linking between nodes as well. The weight of edges are defined by data term and regularization term. Therefore, under certain graph configuration, finding the optimal cutting plan can be equivalent to searching the minimum of energy function Eq. (2.24) according to the third concept. To simplify the explanation, this part focuses on a binary labeling optimization sub-problem constructed on the graph with two nodes $p, q$.

Fig. 2.3 provides an illustration for graph cut. In Fig. 2.3, the edges shown in dot lines are cut, meanwhile the costs are illustrated for each solution. $D$ and $R$ are weight function for edges representing data term and regularization

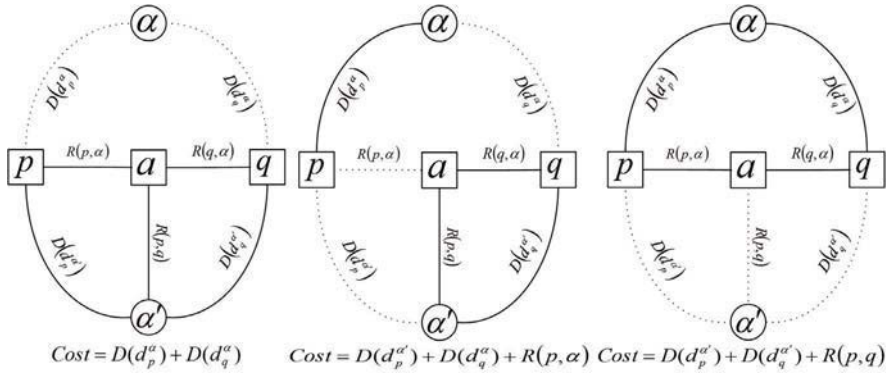



**Figure 2.3:** An illustration of graph cut with three cut solutions

term respectively. The edges linking nodes to terminal nodes are called data term edges, e.g., $d_p^\alpha$ represents the edge linking node $p$ and terminal node $\alpha$, and $p$ has label $d_p$ before relabeling. And the edges linking nodes to other nodes are called regularization term edges, e.g., $(p, q)$ stands for the edge linking between node $p$ and node $q$. For nodes with the label $d_p$ which is different from $\alpha$ before the move, the edges linking them to $\alpha$ are weighted with a data cost retrieved from the cost function. For nodes already having the label $\alpha$, their weights of the edges linking to $\alpha'$ is set to infinity to assure that these nodes keep their $\alpha$-label throughout the optimization. Based on the behavior of cutting data edge explained above, auxiliary nodes (i.e., $a$ in Fig. 2.3) are introduced to model regularization term in the situations which cut data edges linking node and its neighboring nodes to all choices of terminal nodes $\alpha, \alpha'$ (i.e., $p, q$ in Fig. 2.3). The mapping from edge weights in graph to energy function Eq. (2.24) is listed in Tab. 2.1.

With the graph configuration, minimizing energy function equals to solving for the minimum cut of the graph, which is done by computing the maximum flow between the terminals. Maximum flow algorithm is presented in (Ford Jr & Fulkerson 2009).

Graph cuts algorithm is able to solve binary labeling optimization problems precisely. For multi-labeling optimization problem, such as depth estimation and depth enhancement, it only provides approximate solution which

Table 2.1: The Edge Weights in Fig. 2.3

| Edge | Weight | Condition Before Move |
|---|---|---|
| $D\left(d_p^{\alpha'}\right)$ | inf | $d_p = \alpha$ |
| $D\left(d_p^{\alpha'}\right)$ | $data\left(d_p\right)$ | $d_p \neq \alpha$ |
| $D\left(d_p^{\alpha}\right)$ | $data\left(\alpha\right)$ | None |
| $R\left(p,a\right)$ | $\lambda reg\left(d_p,\alpha\right)$ | $q \in \mathbf{N_p}$ |
| $R\left(a,q\right)$ | $\lambda reg\left(\alpha,d_q\right)$ | $q \in \mathbf{N_p}$ |
| $R\left(p,q\right)$ | $\lambda reg\left(d_p,d_q\right)$ | $q \in \mathbf{N_p}$, $d_p \neq d_q$ |
| $R\left(p,q\right)$ | 0 | $q \in \mathbf{N_p}$, $d_p = d_q$ |

is a local minimum close to the global minimum.

## 2.5 Learning-Based Depth SR

In recent years, motivated by common RGB image super-resolution (SR) methods, sparse coding is used for depth map SR. There are two types, synthesis model (Li et al. 2012, Ferstl, Ruther & Bischof 2015, Kwon et al. 2015) and analysis model (Kiechle et al. 2013, Hawe, Kleinsteuber & Diepold 2013). Very recently, some methods based on convolutional neural network (Riegler et al. 2016, Hui et al. 2016) are proposed. Before reviewing the related works, the basic knowledge of sparse coding and convolutional neural network are firstly explained in brief.

### 2.5.1 Sparse Coding

This subsection presents the basic theory of sparse coding and related work based on synthesis model and analysis counterpart.

#### Synthesis Model

One assumption that has proven to be successful in image reconstruction is that natural images admit a sparse representation $\boldsymbol{\alpha} \in \mathbf{R^m}$ over certain

overcomplete dictionary $\mathbf{U} \in \mathbf{R^{n \times m}}$ with $m > n$. A vector $\boldsymbol{\alpha}$ is sparse when most of its coefficients are equal or close to zero. When signal or vectorized image patch $\mathbf{x}$ admits a sparse representation $\boldsymbol{\alpha}$ over $\mathbf{U}$, it can be expressed as a linear combination of only a few columns of the dictionary $\{\mathbf{U}_i\}_{i=1}^{m}$, called atoms, which is shown as Eq. (2.28)

$$\mathbf{x} = \mathbf{U}\boldsymbol{\alpha} \tag{2.28}$$

The dictionary is typically learned via Eq. (2.29).

$$\mathbf{U}^{\star}, \boldsymbol{\alpha}^{\star} = \arg\min_{\mathbf{U},\boldsymbol{\alpha}} \|\mathbf{x} - \mathbf{U}\boldsymbol{\alpha}\|_2^2 \quad subject\ to\ g(\boldsymbol{\alpha}) \leq \epsilon \tag{2.29}$$

where $g$ is a sparsity evaluation function, $\epsilon \in R^{+}$ is an estimated upper bound of sparsity. Ideally, sparsity evaluation function should be 0-norm,i.e., $g(\boldsymbol{\alpha}) = \|\boldsymbol{\alpha}\|_0$. However, since it is difficult to solve, p-norm ($0 < p < 1$) is used instead of 0-norm. In addition to p-norm, there are many other sparsity functions, such as log-square function which is used in (Kiechle et al. 2013). The discussion on them is out of scope of this thesis.

Based on learned dictionary and observed low-quality signal $\mathbf{y}$ degraded by Eq. (2.12), an estimation of the original signal can be obtained by first solving sparse coefficients (i.e., Eq. (2.30))

$$\boldsymbol{\alpha}^{\star} = \arg\min_{\boldsymbol{\alpha}} g(\boldsymbol{\alpha}) \quad subject\ to\ \|\mathbf{H}\mathbf{U}\boldsymbol{\alpha} - \mathbf{y}\|_2^2 \leq \epsilon \tag{2.30}$$

and afterwards synthesizing the signal from the computed sparse coefficients via $\mathbf{x}^{\star} = \mathbf{U}\boldsymbol{\alpha}^{\star}$.

There are representative works for depth map SR based on synthesis model. Li et al. (Li et al. 2012) jointly train dictionaries for registered patches of the LR depth maps, the HR depth maps and the color images. In reconstruction phase, the HR depth maps are reconstructed through sparse representation of learned corresponding dictionary. H. Kwon et.al. (Kwon et al. 2015) proposed a method which takes advantage of a training set of high-quality color and transfer its information to the LR depth map through multi-scale dictionary learning. It learns a dictionary of geometric primitives

which captures the correlation between high-quality mesh data, LR depth maps and HR color images. These two methods share a common form during the learning phase as Eq. (2.31).

$$\mathbf{U}^{\star}, \boldsymbol{\alpha}^{\star} = \arg\min_{\mathbf{U},\boldsymbol{\alpha}} \left\| \begin{bmatrix} \mathbf{x_h} \\ \tilde{\mathbf{x}}_\mathbf{l} \\ \mathbf{x_c} \end{bmatrix} - \begin{bmatrix} \boldsymbol{U}_h \\ \mathbf{U_l} \\ \mathbf{U_c} \end{bmatrix} \boldsymbol{\alpha} \right\|^2 + \lambda \|\boldsymbol{\alpha}\|_1 \tag{2.31}$$

where $\mathbf{x_h}, \tilde{\mathbf{x}}_\mathbf{l}$ and $\mathbf{x_c}$ are features for HR depth patches, LR depth patches and HR color patches respectively. $\mathbf{U_h}, \mathbf{U_l}$ and $\mathbf{U_c}$ are dictionaries for HR depth patches, LR depth patches and HR color patches respectively. $\boldsymbol{\alpha}$ is the sparse coefficients. Overall, H. Kwon et.al. (Kwon et al. 2015) can obtain more robust results than Li et al. (Li et al. 2012). It is an iterative upsampling scheme and introduces Normalized Correlation Coefficient (NCC) measurement for predicting which color edges are most likely to coincide with depth edges. A global reconstruction framework is proposed to mitigate the oversmooth artifacts on the overlapping regions, while Li et al. (Li et al. 2012) reconstruct each patch independently.

### Analysis Model

Different from the synthesis model Eq. (2.28), reconstruction phase for analysis model is to solve Eq. (2.32).

$$\mathbf{x}^{\star} = \arg\min_{\mathbf{x}} g(\boldsymbol{\Omega}\mathbf{x}) \quad \textit{subject to } \|\mathbf{H}\mathbf{x} - \mathbf{y}\|_2^2 \leq \epsilon \tag{2.32}$$

Therein, $\boldsymbol{\Omega} \in \mathbf{R}^{\mathbf{k}\times\mathbf{n}}$ with $k > n$ is called the analysis operator, and the analysed vector $\boldsymbol{\Omega}\mathbf{x} \in \mathbf{R}^{\mathbf{k}}$ is assumed to be sparse, where sparsity is again measured via an appropriate function $g$. In contrast to the synthesis model where a signal is fully described by the nonzero elements of $\boldsymbol{\alpha}$, the zero elements of the analysed vector $\boldsymbol{\Omega}\mathbf{x}$ describe the subspace containing the signal in the analysis model. To emphasize this difference, the term cosparsity is introduced, which simply counts the number of zero elements of $\boldsymbol{\Omega}\mathbf{x}$ (Hawe et al. 2013). The difference between such two types of models

is also shown in the learning phase. Operator learning in analysis model is defined as Eq. (2.33).

$$\mathbf{x}^{\star}, \mathbf{\Omega}^{\star} = \arg\min_{\mathbf{x},\mathbf{\Omega}} g(\mathbf{\Omega}\mathbf{x}) + reg\,(\mathbf{\Omega}) \quad subject\ to\ \|\mathbf{x}-\mathbf{y}\|_2^2 \leq \epsilon \tag{2.33}$$

where $reg\,(\mathbf{\Omega})$ is additional restriction on analysis operator to avoid trivial solutions. Indeed, if no such regularization is imposed on $\mathbf{\Omega}$, it is noticed that the trivial solution $\mathbf{\Omega} \equiv 0$ is the global minimizer. Such regularization on analysis operator is investigated in (Yaghoobi, Nam, Gribonval & Davies 2011, Hawe et al. 2013). Some widely used ones are listed as below;

- row norm constraints; All the rows of $\mathbf{\Omega}$ have the same norm, i.e., $\|\mathbf{\Omega_i}\|_2 = a$ for the i-th row of operator $\mathbf{\Omega}$
- full rank constraints; The analysis operator $\mathbf{\Omega}$ has full rank. i.e., $rank(\mathbf{\Omega}) = n$;
- tight frame constraints; The admissible set of this constraint is the set of tight frame in $\mathbf{R^{k\times n}}$, i.e., $\mathbf{\Omega}^{\top}\mathbf{\Omega} = \mathbf{I_k}$, where $\mathbf{I_k}$ is the identity operator in $\mathbf{R^k}$.

Based on analysis model, Kiechle et al. (Kiechle et al. 2013) exploit the co-sparsity of analysis operators which are applied on depth map and corresponding color image, and reconstructed the HR depth map through data fidelity and color-guided sparsity constraint. The optimized problems in learning and reconstruction phases are shown as Eq. (2.34) and Eq. (2.35) respectively.

$$(\mathbf{\Omega_I^{\star}}, \mathbf{\Omega_D^{\star}}) \in \underset{\mathbf{\Omega_I},\mathbf{\Omega_D}\in \mathrm{OB}(n,k)}{\arg\min} g(\mathbf{\Omega_I}\mathbf{x_I}, \mathbf{\Omega_D}\mathbf{x_D}) + reg(\mathbf{\Omega_I}) + reg(\mathbf{\Omega_D}) \tag{2.34}$$

where $\mathbf{\Omega_I^{\star}}$ and $\mathbf{\Omega_D^{\star}}$ are analysis operators for HR intensity patches and HR depth patches respectively. $\mathbf{x_I}$ and $\mathbf{x_D}$ are registered intensity image patch and the ground truth depth map patch. $g$ is the sparsity function. The analysis operator is restricted to $\mathrm{OB}(n,k)$ which is the set of full-rank matrices

with normalized columns. *reg* states for prior knowledge on such analysis operators. For more details on *reg* please refer to (Kiechle et al. 2013).

$$\mathbf{x}_{\mathbf{D}}^{\star} \in \arg\min_{\mathbf{x_D}} \lambda g(\mathbf{c}, \mathbf{\Omega}_{\boldsymbol{D}}\mathbf{x_D}) + ||\mathbf{H_D x_D} - \mathbf{y_D}||_2^2 \tag{2.35}$$

where $\mathbf{c}$ is a constant which means the analysed intensity signal is not related to $\mathbf{x_D}$. $\mathbf{x_D}$ is HR depth patch, $\mathbf{H_D}$ is the degradation matrix. $\mathbf{y_D}$ is the set of observed depth values.

Although state-of-the-art results can be obtained based on sparse coding, they always need a variety of pre-processing and representation power of these models is inferior to deep convolutional neural network (CNN) which always is an end-to-end model. Next subsection explains CNN briefly and reviews related works based on it.

### 2.5.2 Convolutional Neural Network

Convolutional networks, also known as convolutional neural networks or CNNs, are a specialized kind of neural network for processing data that has a known, grid-like topology. Convolution is a specialized kind of linear operation. Convolutional networks are simply neural networks that use convolution in place of general matrix multiplication in at least one of their layers (Goodfellow, Bengio & Courville 2016).

Depth SR based on convolutional neural network (CNN) are different from sparse coding in which CNN do not explicitly learn dictionaries. Typically, the problems are solved via end-to-end network without specific preprocessing.

At the first attempt, Dong et al. (Dong, Loy, He & Tang 2016) propose an end-to-end super-resolution convolutional neural network (SRCNN) to achieve single image restoration. Subsequently, many single image enhancement methods are proposed based on CNN (Kim, Kwon Lee & Mu Lee 2016, Wang, Liu, Yang, Han & Huang 2015). Compared with single image enhancement, the work based on CNN for guided depth map SR is fewer.

Very recently, Hui et al. (Hui et al. 2016) propose a Multi-Scale Guided convolutional network (MSG-Net) for depth map SR. MSG-Net complements LR depth features with HR intensity features through a multi-scale fusion strategy which progressively resolves ambiguity in depth map SR using the HR intensity features at different levels.

## 2.6 Summary

This chapter firstly addresses basic knowledge for passive depth acquisition (i.e., depth estimation) and active depth acquisition (i.e., depth enhancement). Then the representative techniques (i.e., filters for local methods, graph cut for global optimization and machine learning methods) are introduced including their usage for depth estimation and depth enhancement. In the following chapters, the contributions are explained in details corresponding to which are stated in Introduction section.

# Chapter 3

# Fast Depth Video Construction and Its Enhancement via Temporal Consistency

## 3.1 Related Work and Motivation

Depth maps (depth and disparity is equivalent in this chapter) can be estimated via stereo matching using multi-view color video. An excellent survey of stereo matching methods can be found in Scharstein and Szeliski (Szeliski et al. 2008), where the existing method can be categorized into two categories: local methods and global methods. Local methods have low complexity which compute the depth value for each pixel independently through matching the local information only. Compared with local methods, global methods are more complicated determining the depth values of all the pixels simultaneously based on predefined prior (e.g. the Total Variation norm) (Gong & Yang 2007, Cai 2012, Zhang, Cui, Ngan & Liu 2012, Lei, Selzer & Yang 2006, Ju, Wang & Xiong 2015, Tomioka, Mishiba, Oyamada & Kondo 2016). This chapter focuses on designing a robust local method with low complexity. The related work and corresponding analysis are shown as below.

### 3.1.1 Related Work

Cost aggregation is to generate a volume of which each element is the matching cost between the refer pixel and the target pixel determined by certain disparity candidate. As the explanation in Chapter 2, to be more robust, cost aggregation is performed within a local window for each pixel. To determine such local windows, multiple techniques have been proposed. Multiple window-based methods select the best local window from a set of predefined sizes (Bobick & Intille 1999). Variable-window-based methods do not select the best window from a predefined set, but rather compute an optimal local window for each pixel instead (Veksler 2003). Adaptive-weight-based methods adaptively control the weight for each pixel inside the large fixed-size window (Yoon & Kweon 2006, Hosni, Bleyer, Gelautz & Rhemann 2009). Unlike explicitly determining the shape of each pixel in variable-window methods, adaptive-weight methods implicitly make the decision which introduces the concept of fuzzy.

Among these three types of local methods, adaptive-weight-based methods always reach the best results. The weight represents how likely it shares the similar disparity with the central pixel of the local window. Most adaptive-weight-based methods assume that such weight is defined by the color difference and the pixel location distance between the current pixel and the central pixel (Olgierd Stankiewicz 2009). Therefore, the effort of each pixel within local window in the cost aggregation can be adaptively controlled.

### 3.1.2 Motivation

Despite the relatively satisfy results produced by adaptive-weight-based methods, pixel-wise weight computation and depth estimation are highly time-consuming tasks. They have high computational redundancy because depth maps are always smooth. In this chapter, an adaptive matching scheme is proposed to reduce the computation complexity and to achieve a performance close to or even better than the state-of-the-art local method.

In addition, although most stereo matching methods use color information for matching according to the assumption that the matched pixels should have the same color, it is useless in textureless regions. The lack of visual features makes stereo matching a challenge. To address the matching problem in such regions, motivated by Yang and Ahuja (Yang & Ahuja 2012), the proposed method uses affine invariant feature (AIF) to obtain more robust results. The details are explained in the next section.

## 3.2 Fast Depth Estimation

This section firstly introduces the definition of AIF, then proposes a fast depth estimation method using multi-view color video which includes initial depth estimation and depth refinement.

### 3.2.1 Affine Invariant Feature

#### Definition

Before explain the proposed method, the definition of affine invariant feature (AIF) is addressed as following. There is a straight line $ab$ in Euclidean space $\mathbf{R^3}$. $c$ is a point inside the line $ab$. The line is projected on two planes $\mathbf{L}$ and $\mathbf{R}$ with the corresponding projections $a_L, b_L, c_L$ and $a_R, b_R, c_R$ as shown in Fig. 3.1. $o_L, o_R$ are optical centers of projection planes. The AIF can be modeled as Eq. (3.1) (Liao, Wei & Chen 2007).

$$\frac{c_L - a_L}{b_L - a_L} \equiv \frac{c_R - a_R}{b_R - a_R} \tag{3.1}$$

#### AIF for Depth Estimation

To use AIF for stereo matching, two problems should be discussed firstly. The one is that AIF is theoretically valid for planar surface projection (Hartley & Zisserman 2003). However, the real scene is more complicated. The other one is how to find the two corresponding line segments (e.g., $a_L b_L$ and $a_R b_R$

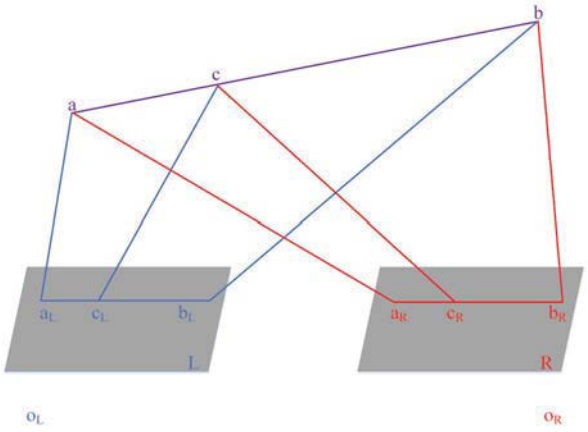



**Figure 3.1:** Illustration of AIF

in Fig. 3.1). The proposed method deals with these two problems as below; (1) It is assumed that depth map consists of planar surfaces as most state-of-the-art stereo matching methods (Ju et al. 2015, Tomioka et al. 2016, Yang & Ahuja 2012, Furukawa, Curless, Seitz & Szeliski 2009). For Lambertian surfaces, higher curvature also means greater variation due to shading and the extent of this intensity variation can be used to partition the whole curved surface into a number of small, low-curvature patches. Each patch can approximate to a planar surface. Therefore, the ratio of the distance, as an AIF, can perform robustly for non-planar, textureless surfaces. (2) It is well known that the two corresponding line segments are in the same row of the scanlines when the image pairs are rectified. In this section, they are extracted by searching along scanlines in left and right directions with a predefined threshold of color differences. Based on the analysis above, the ratio of the line distance can be used as an invariant feature for matching. The detail of computing AIF is explaned as below.

**Computing AIF**

AIF of every pixel can be defined as Eq. (3.2).

$$AIF(p) = \frac{\sum_{q \in \mathbf{L_p}} \omega_q}{\sum_{q \in \mathbf{L_p}+\mathbf{R_p}} \omega_q} \tag{3.2}$$

where $p$ is the current pixel, $\mathbf{L_p}$ and $\mathbf{R_p}$ represents the pixel sets consisting pixels from left bound pixel, right bound pixel of the line segment to $p$ respectively. $q$ is the neighbor pixel of $p$ within this line segment. For every pixel $p$, a global threshold is set to determine $\mathbf{L_p}$ and $\mathbf{R_p}$. The searching along left and right directions are stopped until the luminance absolute difference of two adjacent pixels is larger than predefined threshold. Ideally, $\omega_q$ is the binary value indicating whether this pixel belongs to this line segment or not. To complement with the hard searching threshold for finding line segment, soft-decision scheme is adopted to check the probability of each pixel $q$ belonging to this line segment. The unnormalized probability is modeled as Eq. (3.3).

$$\omega_q = e^{-\frac{(\mathbf{I}(p) - \mathbf{I}(q))^2}{2\delta^2}} \tag{3.3}$$

where $\mathbf{I}(p)$ and $\mathbf{I}(q)$ are luminance of pixel $p, q$. It means that smaller luminance difference represents the higher probability of belonging to the same line segment and vice versa. After computing AIF for all pixels, an AIF image can be obtained which has the same resolution of the color image.

### 3.2.2 Initial Depth Estimation

This part proposes the method for initial depth estimation. Among the existing methods, there is a common problem for cost aggregation which is performed within local windows; on the one hand, it is hard to provide enough features by adopting a small fixed window in textureless regions which makes matching a challenge, on the other hand, it leads to inaccurate depth edges

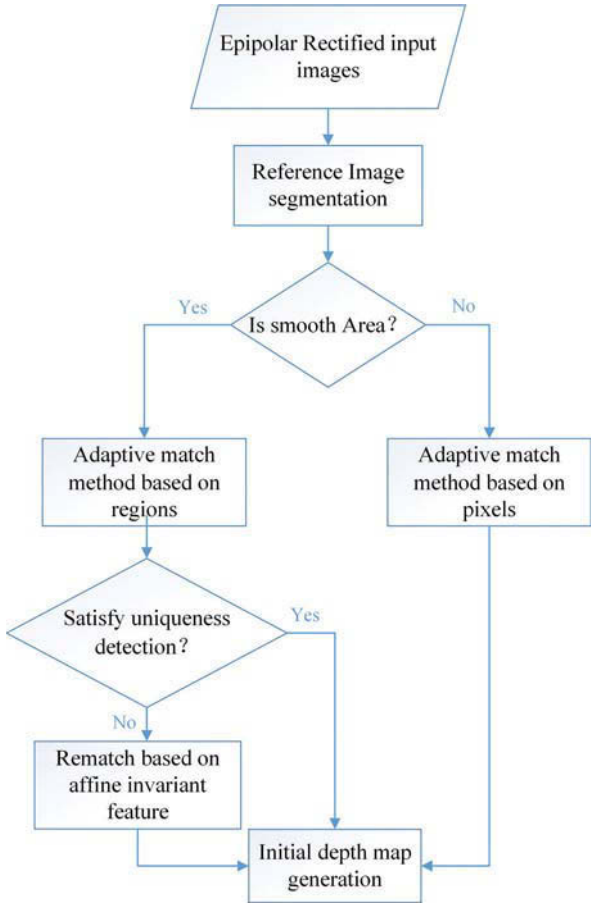



**Figure 3.2:** The flowchart of initial depth estimation

by using a large window in high-texture regions. The proposed adaptive matching scheme classifies the situations which are described above. Specially, for textureless regions, the adaptive windows are generated based on over-segmentation instead of time-consuming methods (Hosni et al. 2009, Olgierd Stankiewicz 2009). The proposed method obtains a reference depth value for the whole segment, followed by the refinement for each pixel within a small range of disparity candidates. Such processing leads to pixels which have similar color gathering in an adaptive local window which helps deal with the matching problem in textureless regions. For high-texture regions, to obtain the accurate edges and reduce the running time, a small local win-

dow is adopted. Thanks to the adaptive matching scheme above, the proposed depth estimation method can reduce the running time compared with the state-of-the-art local depth estimation methods. Fig. 3.2 is the flowchart of the proposed method which is described in detail in the following parts.

### Initial Cost Aggregation

The first step for depth estimation is cost aggregation which is performed via proposed adaptive matching scheme. The proposed method is to obtain the registered disparity map for the color image under certain view which is called reference image, and the ones under other views are called target images.

The reference image is over-segmented into regions through mean-shift segmentation algorithm (Comaniciu & Meer 2002). All the regions constitute the set $\boldsymbol{\pi}(\pi_1, ...\pi_{max})$. Based on the amount of pixels in each region $Num(\pi_i)$, $\pi_i$ can be classified into smooth region when $Num(\pi_i)$ is greater than the predefined threshold, otherwise it is classified into unsmooth region. For smooth regions, due to the over-segmentation, it is more likely to satisfy the assumption that the disparities of pixels in each region are similar. Therefore, simple sum of absolute differences measure (SAD) is used to obtain the matching cost under certain disparity value sharing between all pixels in this region, which is shown as Eq. (3.4).

$$SAD(k,d) = \sum_{p \in \pi_k} |\mathbf{I_{ref}}(p) - \mathbf{I_{tar}}(p+d)| \tag{3.4}$$

where $SAD(k,d)$ represents the matching error of the $k-th$ region under the disparity candidate $d$. $\mathbf{I}$ is the illumination of certain image. $p+d$ denotes the target pixels in the target image corresponding to pixel $p$ in the reference image.

For unsmooth regions, cost aggregation is performed for every pixel within a local window. By considering the computational complexity, soft segmentation which is introduced in Section 2.3 is only used for the reference local win-

dow. To make the section content integrity, the method is illustrated briefly as follows. Eq. (3.5) is used to calculate the adaptive weights $W_{ref}$ for pixels in the reference local window. Such weights are embedded into Eq. (3.6) to compute weighted mean of absolute intensity differences (WMAD). The size of the matching window is 5×5.

$$\mathbf{W_{ref}}\left(p,p'\right)=e^{-\frac{\left|\mathbf{I_{ref}}(p)-\mathbf{I_{ref}}(p')\right|}{\gamma_c}-\frac{|p-p'|}{\gamma_d}} \tag{3.5}$$

$$WMAD\left(p,d\right)=\frac{\sum\limits_{p'\in\mathbf{N_p}}\mathbf{W_{ref}}\left(p,p'\right)\times\left|\mathbf{I_{ref}}\left(p\right)-\mathbf{I_{tar}}\left(p+d\right)\right|}{\mathbf{W_{ref}}\left(p,p'\right)} \tag{3.6}$$

where $p'$ is the pixel in the local window $\mathbf{N_p}$ centered at pixel $p$ in the reference image. $\gamma_c, \gamma_d$ are bandwidths for range kernel and distance kernel respectively.

To prevent generating wrong depth edges, the proposed method adopts a complex matching measure consisting of WMAD and the weighted mean of gradient absolutely difference (WMGRAD). $\nabla_h, \nabla_v$ are the horizontal and vertical gradient of intensity image respectively. The WMGRAD is described as Eq. (3.7) with the same symbols defined in Eq. (3.6).

$$\begin{aligned}
WMGRAD_h\left(p,d\right)&=\frac{\sum\limits_{p'\in\mathbf{N_p}}\mathbf{W_{ref}}\left(p,p'\right)\times\left|\nabla_h^{ref}\left(p\right)-\nabla_h^{tar}\left(p+d\right)\right|}{\mathbf{W_{ref}}\left(p,p'\right)}\\
WMGRAD_v\left(p,d\right)&=\frac{\sum\limits_{p'\in\mathbf{N_p}}\mathbf{W_{ref}}\left(p,p'\right)\times\left|\nabla_v^{ref}\left(p\right)-\nabla_v^{tar}\left(p+d\right)\right|}{\mathbf{W_{ref}}\left(p,p'\right)}
\end{aligned} \tag{3.7}$$

In addition, the proposed method adaptively controls the effects of WMAD and WMGRAD in matching cost computing. As the WMAD is not robust when the illumination differences of two images (i.e., the reference image and the target image) is significant, the efforts of it is set to be inversely proportional to the average of illumination differences in this method which

is shown in Eq. (3.8).

$$\Omega_{comb} = \frac{1}{1 + \alpha \times md} \tag{3.8}$$

where $md$ represents the mean of illumination difference between the registered images, $\alpha$ defines the weight of $md$.

When depth sequence is estimated frame by frame, temporal consistency cannot be guaranteed. Since the depth values in background should not change between two adjacent frames, it is necessary to add the temporal consistency term $temp$ to the cost aggregation as shown in (Sang-Beom Lee & Ho n.d.). This method calculates mean absolute difference (MAD) for each block between adjacent frames and determines whether the block is background according to threshold. The temporal consistency term is defined as Eq. (3.9) and Eq. (3.10).

$$temp\left(p, d^t, d^{t-1}\right) = \lambda^t \left|d^t - d^{t-1}\right| \tag{3.9}$$

$$\lambda^t = \begin{cases} 1, & if\ p \in\ background \\ 0, & otherwise \end{cases} \tag{3.10}$$

where $d^t$ and $d^{t-1}$ represent disparities of certain pixel for current frame and the previous frame respectively.

Based on the analysis above, the final cost volume is constructed as Eq. (3.11) and Eq. (3.12) according to smooth regions and unsmooth regions.

$$Cost\left(k, d^t\right) = SAD\left(k, d^t\right) + \sum_{p \in \pi_k} temp\left(p, d^t, d^{t-1}\right) \tag{3.11}$$

$$\begin{aligned} Cost\left(p, d^t\right) =& \Omega_{comb} \times WMAD\left(p, d^t\right) + (1 - \Omega_{comb}) \\ & \times WMGRAD\left(p, d^t\right) + temp\left(p, d^t, d^{t-1}\right) \end{aligned} \tag{3.12}$$

### Cost Volume Refinement

Although over-segmentation can make the assumption that pixels in each segment have similar disparities more likely to be satisfied, a large smooth region may be partitioned into several smaller regions. In such situation, the matching problem for textureless regions cannot be solved. The cost volume of smooth regions and pixels in unsmooth regions are unreliable if they do not satisfy uniqueness test (i.e., Eq. (3.13)).

$$\frac{C^2_{opt} - C^1_{opt}}{C^1_{opt}} > T \tag{3.13}$$

where $C^1_{opt}$ and $C^2_{opt}$ are the optimal value and the sub-optimal value in the cost volume.

To prevent the mismatch for whole smooth region, the proposed method matches the unreliable smooth regions again on AIF images through Normalized Cross-Correlation (NCC) (Zhang, Lu, Lafruit, Lauwereins & Van Gool 2009) instead of SAD. If the rematch is satisfy Eq. (3.13), the recomputing costs in disparity range for the unreliable smooth region are updated to the co-location in cost volume. And this unreliable smooth region is reset to be the reliable region.

### Initial Depth Generation

After constructing cost volume, Winner-Take-All (WTA) method is adopted to locally select the best disparity $d_p$ for certain pixel $p$ in unsmooth regions and $d_k$ for the super-pixel $k$ in smooth regions which are shown as Eq. (3.14) and Eq. (3.15).

$$d_k = \begin{cases} \underset{d\in[d_{min},d_{max}]}{\arg\max} \; Cost\left(k, d^t\right), & if\ Cost(k, d^t)\ is\ updated \\ \underset{d\in[d_{min},d_{max}]}{\arg\min} \; Cost\left(k, d^t\right), & otherwise \end{cases} \tag{3.14}$$

$$d_p = \underset{d\in[d_{min},d_{max}]}{\arg\min} \; Cost\left(p, d^t\right) \tag{3.15}$$

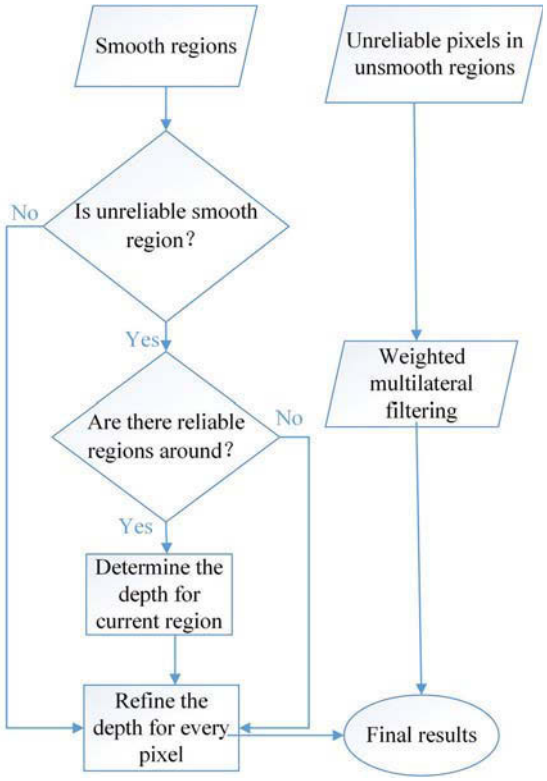



**Figure 3.3:** The flowchart of refinement for initial depth

### 3.2.3 Depth Map Refinement

Since there are errors left in the initial depth maps, the special refinement is proposed for smooth regions and unsmooth regions respectively. Fig. 3.3 is the flowchart.

#### Refinement for Smooth Regions

Owing to over-segmentation, the adjacent smooth regions may have the similar reliable disparities. Based on this assumption, the unreliable smooth regions can be refined via the adjacent reliable smooth regions.

The proposed method searches within the neighbor of each unreliable smooth region, and puts all reliable smooth regions into a candidate list. If the mean of RGB color channels of a region $m(\pi_i)$ in the list is quite

different from that of current unreliable smooth region $m(\pi_{curr})$, the region $\pi_i$ will be deleted from the list. Then, the proposed method updates the depth of current smooth region using depth of the region in the list whose color is closest to it. If there are no sufficient candidates in the list, the depth value of current region is unchanged.

Since the disparities of pixels in every smooth region are similar but not completely consistent with each other, it is necessary to fine tune disparity for every pixel in smooth regions. A narrow searching range is determined for every pixel in the smooth regions based on current reference disparities. Although the matching measure based on color performs poor in the textureless region, the AIF features can distinguish pixels whose colors are similar in textureless regions. Therefore, fine tune is performed on AIF images based on the NCC matching measure with a $5 \times 5$ local window. In the proposed method, the disparity range is chosen as $d \in [d_{ref} - 2, d_{ref} + 2]$. For robustness, the depth value of the corresponding pixel is updated, only when the maximum value is larger than 0.85, and it satisfies Eq. (3.13).

#### Refinement for Unsmooth Regions

For unreliable pixels in unsmooth regions, the depth values of them are refined by using the multi-lateral filtering (Lo, Wang & Hua 2013). In the filtering process, the weights of pixels which belong to the unreliable smooth regions and unreliable pixels in unsmooth regions are set to zero. Since the processing is just executed for the unreliable pixels, little computational complexity is added.

### 3.2.4 Experimental Results

In this section, the proposed method is tested on three sequences which are "Akko", "Lovebird2" and "Book arrival", whose resolution are $640 \times 480, 720 \times 576$ and $1024 \times 768$ respectively. All the test sequences are 100 frames in the experiments. The results of the proposed method with/without

Table 3.1: Quantitative Evaluation of Initial Results on Average PSNR of Rendered Color Images

| Methods / Sequences | FDE | FDE w/o AIF | DERS w/o GC |
|---|---|---|---|
| Akko | 31.81 | 31.48 | 31.29 |
| Lovebird2 | 31.43 | 31.33 | 30.87 |
| Book arrival | 34.81 | 34.52 | 34.91 |

refinement and DERS without Graph cut (DERS w/o GC) (Olgierd Stankiewicz 2009) are shown.

**Objective Evaluate for Initial Results**

To objectively evaluate the initial depth maps estimated by the proposed fast depth estimation (FDE), the average peak signal to noise ratio (PSNR) of rendered virtual images are shown in Tab. 3.1 comparing with results of the propose method without AIF (FDE w/o AIF) and DERS w/o GC. From the table, it is shown that FDE provides highest average PSNR for "Akko" and "Lovebird2". In addition, the PSNR for every frame for three methods above is shown in Fig. 3.4 which shows that FDE reaches the best performance for 'Akko' and 'Lovebird2' and provides comparable result for 'Book arrival' with DERS w/o GC.

**Subjective Evaluate for Initial Results**

In this part, a subjective evaluation on the proposed method is given followed by a detailed analysis. Fig. 3.5 shows the depth maps estimated by the proposed fast depth estimation (FDE), FDE without using AIF images (FDE w/o AIF) and DERS w/o GC. From the highlighted regions, it can be seen that the depth maps estimated by FDE seem perceptually more close to the scene. The first row shows the results of 'Akko', where FDE gives more accurate edge information, especially at positions of the left woman and ball. The

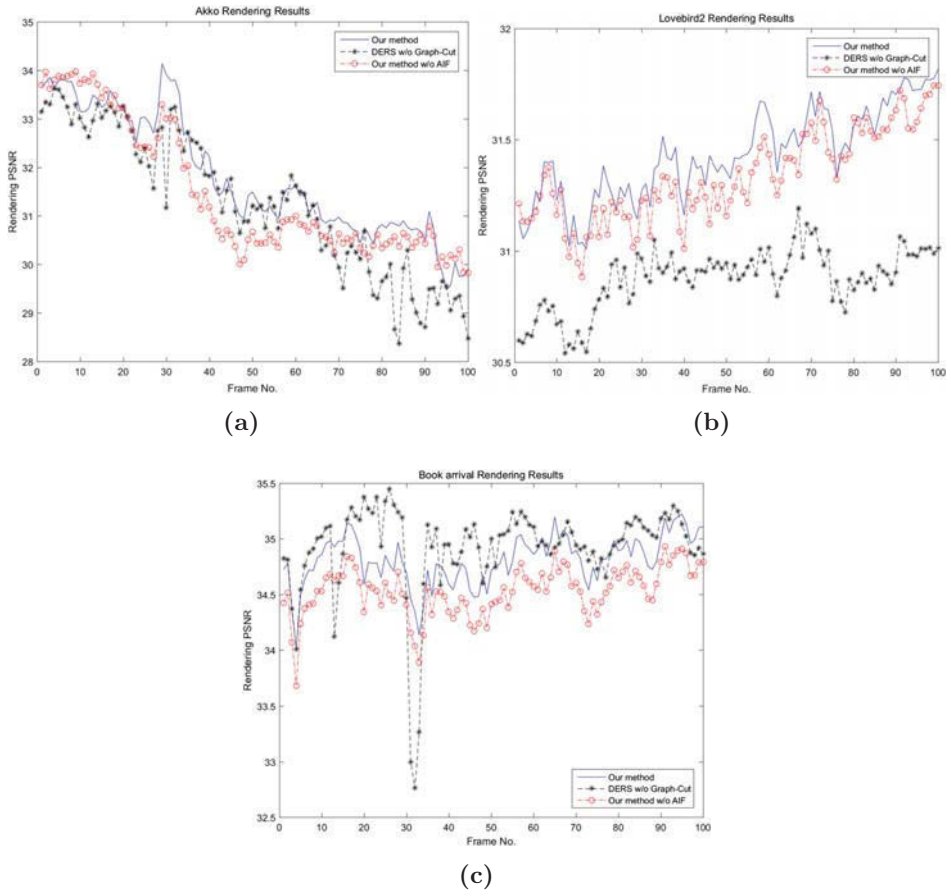
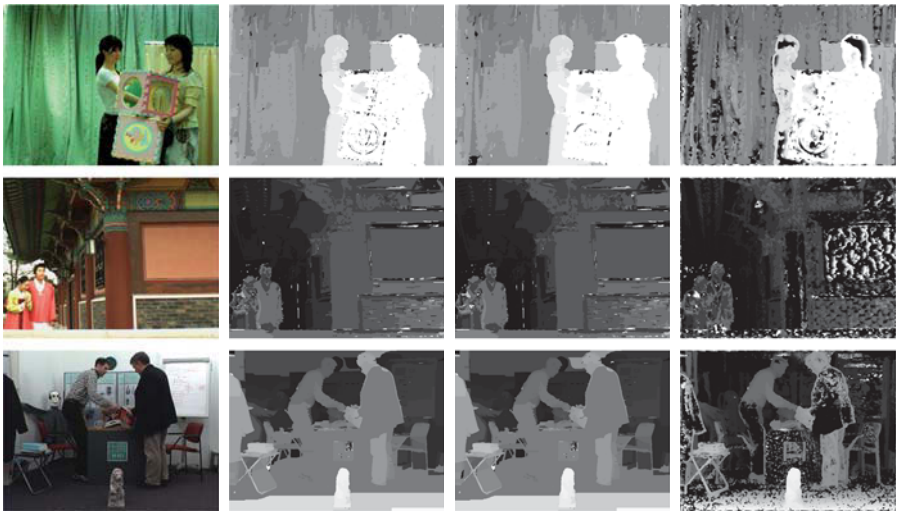



**Figure 3.4:** Objective evaluate for initial estimated results: (a) rendering result of Akko, (b) rendering result of Lovebird2, (c) rendering result of Book arrival.

results of DERS w/o GC (Olgierd Stankiewicz 2009) are rather poor in the textureless regions. When the AIF is used, some unreliable smooth regions can be corrected. In addition, the depth of blackboard area in the second row estimated by FDE is basically true. But the depth accuracy in this region estimated by DERS w/o GC (Olgierd Stankiewicz 2009) is relatively lower. To combine the objective evaluation with subjective evaluation for "Akko" and "Lovebird2", it can be noticed that the results of the proposed method obtain the best performance.

For "Book arrival", although FDE obtains the best result in most fore-



(a) (b) (c) (d)

**Figure 3.5:** Subjective evaluation for initial results: (a) color images, depth map estimated by (b) FDE, (c) FDE w/o AIF, (d) BAW.

ground, the depth values of background are wrongly estimated, especially at the floor in the bottom of the scene. Comparing with the results of FDE w/o AIF, the improvement of using the AIF image is shown, especially near the right mans head.

The reason why the depth values of background are wrongly estimated in FDE is that the scene is inclining. In this situation, although the color is similar in the floor region, the disparities of the pixels in those regions vary from each other along the inclining plane. Therefore, the matching of floor region is unreliable.

### Experimental Results of Refinement

To prove the validity of the refinement, the PSNR of every rendered virtual frame by using refined depth map is given which are shown in Fig. 3.6. The average PSNR of them are given in Tab. 3.2. For the same reason as initial depth estimation, the refinement of "Book arrival" is not significant

Table 3.2: Quantitative Evaluation of Refinement on Average PSNR of Rendered Color Images

| Depth / Sequences | Initial Results | Refinement Results |
|---|---|---|
| Akko | 31.81 | 31.93 |
| Lovebird2 | 31.43 | 31.56 |
| Book arrival | 34.81 | 34.86 |

compared with other sequences.

**Complexity Analysis**

For all pixel-wise estimation methods, their computational complexity can be denoted by $O(TS_dS_w)$, where $S_d$ and $S_w$ are the disparity range and the fixed size of the matching window. $T$ is the resolution of color image. By contrast, the proposed method uses adaptive matching scheme. All pixels in the same smooth region get a same reference disparity, and then a small range of disparity candidates is adopted to refine the disparity for every pixel. In high-texture regions, the small window size $S'_w < S_w$ is enough to get the satisfactory results. Based on the analysis above, the computational complexity of the proposed method can be denoted by $O(TS'_dS'_w)$, where $S'_d < S_d$ is the average searching range of all pixels. Furthermore, since the number of unreliable regions and pixels are small compared with the original resolution of the images, the computation of refinement is little. In fact, most of the running time is consumed in the reference image segmentation which is shown in Tab. 3.3.

The experiments environment is a PC (i3 Intel processor with 2 GB RAM). The total average running time and the running time of all parts including segmentation (Segment), cost aggregation (Matching) and refinement (Refine) are given in Tab. 3.3. It is shown that the computational complexity of the proposed method is lower than Stankiewicz (Olgierd Stankiewicz 2009).

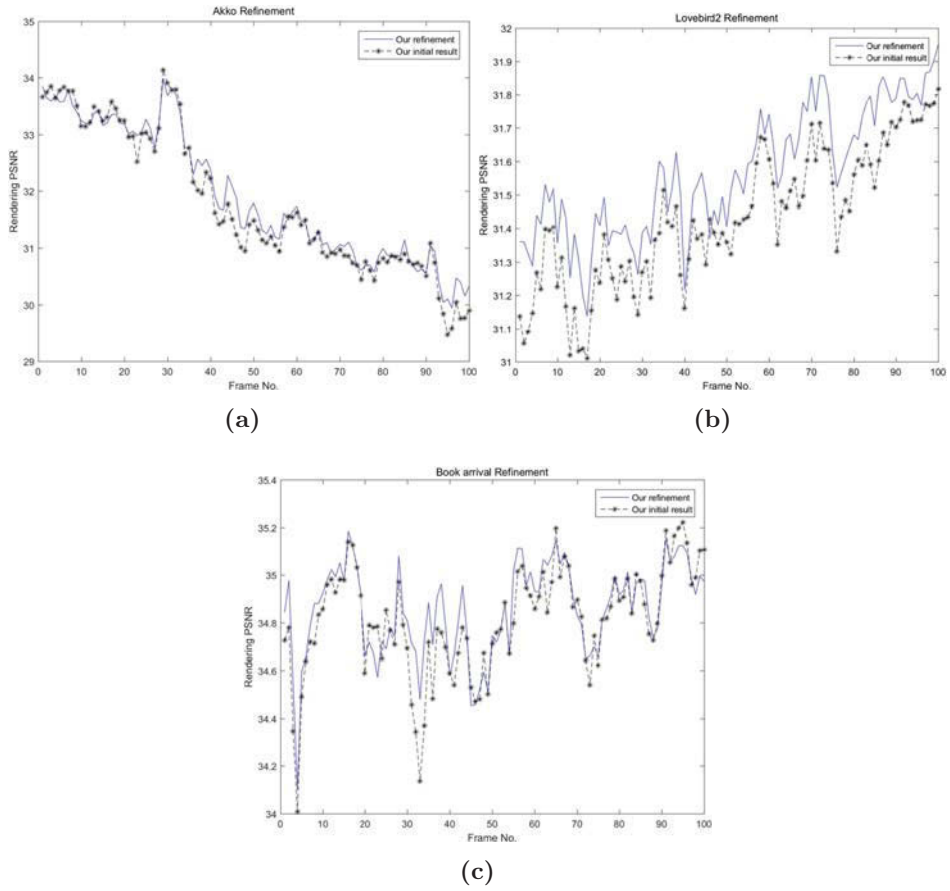



**Figure 3.6:** Objective evaluate for refinement: (a) rendering result of Akko, (b) rendering result of Lovebird2, (c) rendering result of Book arrival.

### 3.2.5 Conclusion

A fast and effective depth estimation method for multi-view video is proposed by using adaptive matching scheme and AIF images. Three sequences are used to validate the presented method. The results show that it not only provides robust depth maps but also reduces running time.

Table 3.3: Average Running Time Comparison

| Methods / Sequences | DERS w/o GC | FDE | | | |
|---|---|---|---|---|---|
| | | Segment | Matching | Refine | Total |
| Akko | 55s | 20s | 6s | 0.4s | 26.4s |
| Lovebird2 | 60s | 25s | 7s | 0.6s | 32.6s |
| Book arrival | 280s | 75s | 18s | 3s | 96s |

## 3.3 Temporal Consistency Enhancement for Multi-view Depth Sequences

Since depth maps are estimated frame by frame, temporal consistency problem occur not only in results of local methods, but also global methods. It leads to flashing artifacts when rendering virtual view using such depth maps. The reason is that the depth values of certain regions which should have had the same depth along adjacent frames are different with obvious gaps. In addition, it reduces the compress efficiency of depth sequence which have to allocate more bit to save such artificial inconsistency. This section proposes a temporal enhancement method for multi-view depth sequences.

### 3.3.1 Motivation

Although DERS (Sang-Beom Lee & Ho n.d.) performs estimation by designing a temporal term in energy function, such term is difficult to balance with other terms among frames which leads to unstable results. Fu et.al. (Fu et al. 2010) proposed a method averaging the static scene of adjacent frames to enhance temporal consistency. However, it does not consider the reliability of depth values participating in average which contributes to error propagation. Based on the analysis above, this section proposes a temporal consistency enhancement method for depth sequences explicitly considering reliability of depth and moving attribute of regions.

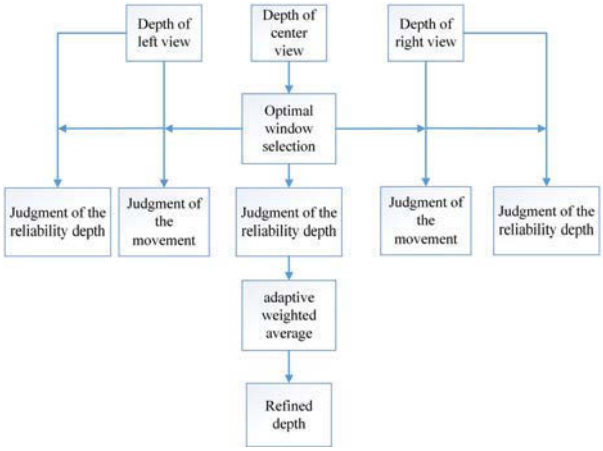



**Figure 3.7:** Framework of temporal consistency enhancement

### 3.3.2 Proposed Method

This subsection illustrates the proposed method in detail. The flowchart is shown as Fig. 3.7. The proposed adaptive weight in weighted average consists of two parts; 1. probability of belonging to static scene, 2. reliability of depth values. By considering probability of belonging to static scene for pixels, it can mitigate error propagation to moving regions. And reliability of depth values can reduce influence of unreliable depth in weighted average which mitigate error propagation among adjacent frames. At the same time, because it is assumed that sequence can be modeled as Markov model, the previous and next frames have the strongest relationship with the frame to be enhanced. Therefore, the proposed weighted average is restricted to two adjacent frames which is designed as Eq. (3.16). To simplify equation, all the depth values $d$ in certain depth frame to be enhanced are gathered into $\mathbf{D}$. And in Eq. (3.16), all the pixels are element-wise processing, but not matrix multiplication.

$$\mathbf{D}'_{\mathbf{curr}} = \frac{\boldsymbol{\omega}^{\boldsymbol{p}}\boldsymbol{\beta}^{\boldsymbol{p}}\mathbf{D}'_{\mathbf{prev}} + \boldsymbol{\beta}^{\boldsymbol{c}}\mathbf{D}_{\mathbf{curr}} + \boldsymbol{\omega}^{\boldsymbol{n}}\boldsymbol{\beta}^{\boldsymbol{n}}\mathbf{D}_{\mathbf{next}}}{\boldsymbol{\omega}^{\boldsymbol{p}}\boldsymbol{\beta}^{\boldsymbol{p}} + \boldsymbol{\beta}^{\boldsymbol{c}} + \boldsymbol{\omega}^{\boldsymbol{n}}\boldsymbol{\beta}^{\boldsymbol{n}}} \tag{3.16}$$

where $\mathbf{D}_{\mathbf{curr}}, \mathbf{D}_{\mathbf{prev}}$ and $\mathbf{D}_{\mathbf{next}}$ are depth values of current depth frame, previous depth frame and next depth frame respectively. $\mathbf{D}'$ is depth values

of enhanced depth frame. $\boldsymbol{\omega^p}$ and $\boldsymbol{\omega^n}$ are probability of belonging to static scene for all pixels in previous depth frame and next depth frame. $\boldsymbol{\beta^p}, \boldsymbol{\beta^c}$ and $\boldsymbol{\beta^n}$ are reliability of depth values for all pixels in previous depth frame, current depth frame and next depth frame respectively. In the rest parts of this subsection, the details of $\boldsymbol{\omega}$ and $\boldsymbol{\beta}$ computing are addressed.

### Computing Probability of Belonging to Static Scene

Based on the observation that static regions share the similar color between adjacent frames, Mean of Absolute Differences (MAD) is used to evaluate whether pixels belong to static regions. The evaluation is performed within a local window centered at the pixel to be evaluated. Motivated by bilateral filter (Tomasi & Manduchi 1998), the reliability of static for pixel $i$ is modeled as exponential function Eq. (3.17). To be robust, such weight is truncated to 0 for pixels belonging to moving regions with large MAD.

$$\boldsymbol{\omega}(i) = \begin{cases} e^{\frac{-MAD_\omega\left(\mathbf{ref}_\omega^\mathbf{i}, \mathbf{tar}_\omega^\mathbf{i}\right)}{\delta_\omega}}, & MAD\left(\mathbf{ref}_\omega^\mathbf{i}, \mathbf{tar}_\omega^\mathbf{i}\right) < Th_{static} \\ 0, & MAD\left(\mathbf{ref}_\omega^\mathbf{i}, \mathbf{tar}_\omega^\mathbf{i}\right) \geqslant Th_{static} \end{cases} \tag{3.17}$$

where $\boldsymbol{\omega}(i)$ is an element of $\boldsymbol{\omega}$ representing the weight of i-th pixel in the depth frame. $\mathbf{ref}_\omega^\mathbf{i}$ and $\mathbf{tar}_\omega^\mathbf{i}$ are color of pixels within local windows centered at $i$ in adjacent two depth frames. So $\boldsymbol{\omega^p}$ is computed between current depth frame and previous depth frame. And $\boldsymbol{\omega^n}$ is computed between current depth frame and next depth frame. $Th_{static}$ and $\delta_\omega$ are predefined parameters.

### Computing Reliability of Depth

Due to noise in estimated depth maps, reliability of depth values should be explicitly considered which gives larger weight to more reliable depth. For each pixel in depth map of current view, since the corresponding pixel in other view can be computed by using disparity, the reliability of depth can be evaluated by the MAD between the local windows centered at such pixel pair. To handle occlusion, the evaluation is performed on left and right

adjacent views and lower MAD is chosen. Similar to computing probability of belonging to static scene, the reliability of depth for pixel $i$ is modeled as exponential function as well. The equation is shown as Eq. (3.18) and Eq. (3.19).

$$\boldsymbol{\beta}(i) = e^{\frac{-MAD^i}{\delta_\beta}} \tag{3.18}$$

$$MAD^i = min\left(MAD\left(\mathbf{ref}_{\boldsymbol{\beta}}^{\mathbf{i}}, \mathbf{left}_{\boldsymbol{\beta}}^{\mathbf{i+d_i}}\right), MAD\left(\mathbf{ref}_{\boldsymbol{\beta}}^{\mathbf{i}}, \mathbf{right}_{\boldsymbol{\beta}}^{\mathbf{i-d_i}}\right)\right) \tag{3.19}$$

where $\boldsymbol{\beta}(i)$ is an element of $\boldsymbol{\beta}$ representing the reliability of i-th pixel value in the depth frame. $d_i$ is the disparity for $i$. Therefore, the corresponding pixels for $i$ in current view are $i + d_i$ and $i - d_i$ in the left and right views respectively. $\mathbf{ref}_{\boldsymbol{\beta}}^{\mathbf{i}}$ and is the local windows centered at $i$ in current view. $left$ and $right$ are depth maps in left and right adjacent views. $\delta_\beta$ is a predefined parameter.

### Window Size Selection

Since probability of belonging to static scene and reliability of depth values are computed within a local window, it is necessary to adaptively select the size of such windows. For computing reliability of depth, it requires as many pixels as possible in local window sharing the same depth. So a small window size is preferred. However, evaluation within a large window is more robust to noise. In the proposed method, a balanced scheme is designed which predefines a list of window sizes including $9 \times 9$, $7 \times 7$, $5 \times 5$ and $3 \times 3$.

The largest size is firstly selected, and the variance $var$ for such local window in the depth map is computed. If $var$ is larger than the predefined threshold, it represents that the depth values within local window are not similar. So it should reduce the window size and recheck $var$ in the new window. The processing is stopped either $var$ is less than threshold or the smallest size is selected.

(a) (b) (c)

**Figure 3.8:** Static regions of test sequences for evaluation marked by black rectangles, (a) static region for Akko, (b) static region for Lovebird2, (c) static region for Newspaper

### 3.3.3 Experimental Results

This subsection includes four parts: 1. temporal consistency evaluation on enhanced depth maps directly, 2. PSNR on rendered color images, 3. coding efficiency for enhanced depth maps, 4. subjective evaluation of temporal consistency. The proposed method is tested on "Akko", "Lovebird2" and "Newspaper" sequences each of which has five views and 100 frames for each view. All initial depth sequences are estimated by (Sang-Beom Lee & Ho n.d.) (DERS). Fu et.al. (Fu et al. 2010) (Fu) and the proposed method (Pro) are post-processing for such initial depth sequences.

#### Temporal Consistency Evaluation

Temporal consistency is evaluated on the marked regions shown as Fig. 3.8 which are static scenes. In such regions, depth of pixels on the same location within all frames are gather as a random variable. The variance of each variable is computed independently. The mean of all variances $\varphi$ is the metric for evaluation.

Tab. 3.4 shows the result for temporal consistency enhancement which includes two views. From Tab. 3.4, the smallest $\varphi$ is shown in the proposed method which proves its validity.

Table 3.4: Consistency Evaluation for Static Regions

| Methods / Sequences | DERS | Fu | Pro |
|---|---|---|---|
| Akko | view 27: 4.13783 | view 27: 2.97219 | view 27: 2.16260 |
| | view 29: 4.23303 | view 29: 2.88039 | view 29: 2.18859 |
| Lovebird2 | view 8: 5.74500 | view 8: 4.10575 | view 8: 3.29371 |
| | view 10: 1.02452 | view 10: 0.32442 | view 10: 0.02412 |
| Newspaper | view 4: 2.68074 | view 4: 1.50024 | view 4: 1.49006 |
| | view 6: 3.10664 | view 6: 1.8965 | view 6: 1.88509 |

Table 3.5: Comparison of Average PSNR of Rendered Color Images

| Methods / Sequences | DERS | Fu | Pro |
|---|---|---|---|
| Akko | 33.70 dB | 33.73 dB | 33.75 dB |
| Lovebird2 | 31.65 dB | 31.70 dB | 31.72 dB |
| Newspaper | 32.67 dB | 32.69 dB | 32.74 dB |

### PSNR on Rendered Color Images

To further validate the proposed method, the quality of rendered color images by using enhanced depth maps are evaluated. Tab. 3.5 shows average PSNR for rendered color images. It presents that the quality of depth maps are improved by the proposed method.

### Coding Efficiency Evaluation

This part shows the bit rate (BR) of coding depth sequence enhanced by the proposed method as well as average PSNR of rendered color images using corresponding decoded depth maps. Testing sequences are “Akko”, “Lovebird2” and “Newpaper” with three QPs (i.e., 22, 27 and 32) for each sequence. Tab. 3.6 provides the performance of the proposed method (Pro)

Table 3.6: Comparison of Depth Sequence Coding Performance

| Sequences \ Methods | QPs | BR-DERS kbit/s | PSNR-DERS dB | BR-Pro kbit/s | PSNR-Pro dB | ∇ BR % | ∇ PSNR dB |
|---|---|---|---|---|---|---|---|
| Akko | 22 | 762 | 34.27 | 668 | 34.36 | -12.3 | +0.09 |
| | 27 | 428 | 34.33 | 341 | 34.39 | -20.3 | +0.06 |
| | 32 | 227 | 34.32 | 176 | 34.40 | -22.4 | +0.08 |
| Lovebird2 | 22 | 1559 | 31.57 | 1141 | 31.65 | -26.8 | +0.08 |
| | 27 | 858 | 31.63 | 585 | 31.69 | -31.8 | +0.06 |
| | 32 | 433 | 31.67 | 285 | 31.74 | -34.2 | +0.07 |
| Newspaper | 22 | 429 | 32.57 | 354 | 32.58 | -17.5 | +0.01 |
| | 27 | 228 | 32.62 | 191 | 32.63 | -16.2 | +0.01 |
| | 32 | 120 | 32.64 | 102 | 32.64 | -15.0 | 0 |

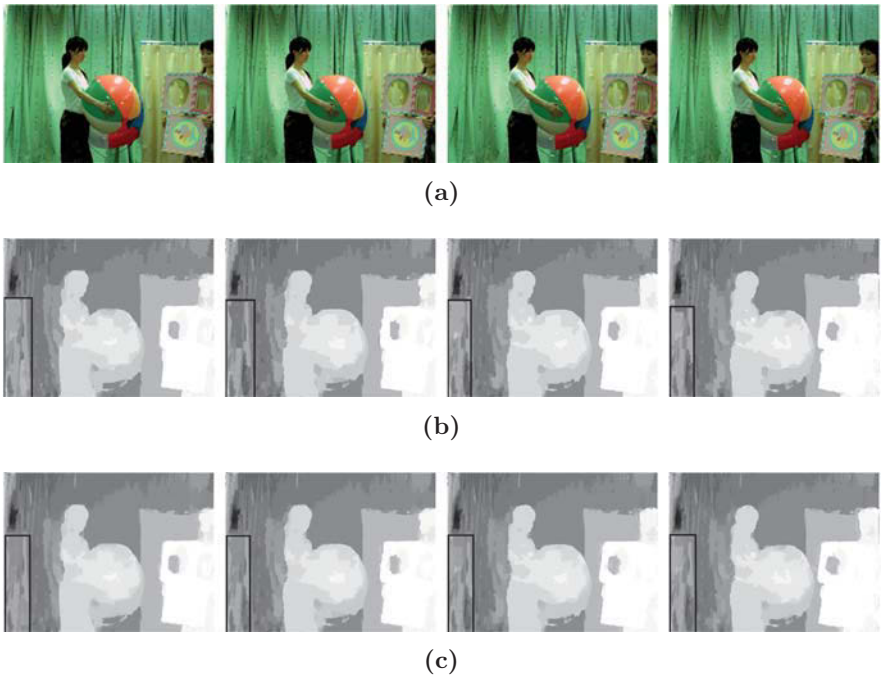
comparing with (Sang-Beom Lee & Ho n.d.) (DERS). It illustrates that the depth sequence enhanced by the proposed method not only significantly saves bit-rate, but also improves the quality of rendering color images.

**Subjective Evaluation of Temporal Consistency Enhancement**

Since the proposed method only enhances the depth for static regions to mitigate flashing artifacts in rendered images, the average PSNR does not gain significant improvement. However, the improving temporal consistency of enhanced depth sequence can be seen via subject evaluation as shown in Fig. 3.9 which places multiple adjacent depth maps for subjective evaluation.

### 3.3.4 Conclusion

This section proposes a temporal enhancement method for estimated multi-view depth sequences. It is an adaptive weighted average scheme which explicitly considers reliability of depth and moving attribute of regions. Experimental results proves that the coding efficient for enhanced depth maps can be significantly improved. Since the proposed method only enhances the depth for static regions to mitigate flashing artifacts in rendering virtual images, the average PSNR does gain significant improvement. However, subjective evaluation shows that the temporal consistency of depth sequence is improved.



**(a)**

**(b)**

**(c)**

**Figure 3.9:** Subjective evaluation of temporal consistency, (a) 4 consecutive frames of color sequence (b) corresponding 4 consecutive depth maps estimated by DERS (c) corresponding 4 consecutive depth maps enhanced by the proposed method

## 3.4 Summary

This chapter firstly proposes a passive acquisition method via fast stereo matching which takes adaptive matching scheme and affine invariant feature into account. It not only decreases computing complexity, but also obtains more robust results on textureless regions. Subsequently, a temporal enhancement method is proposed for estimated multi-view depth sequences to mitigate flashing artifacts in rendering virtual images. Such two methods are validated by sufficient experiments.

# Chapter 4

# Color-Guided Depth Map Enhancement for RGB-D Data via Markov Random Field

## 4.1 Related Work

Depth enhancement consists of two different problems which are depth map super-resolution (SR) and depth map completion according to ToF sensors and structured-light sensors. Depth map SR focuses on enlarging the spatial resolution and depth completion focuses on filling the missing depth values. Such two problems are similar and can be casted into a uniform formulation (Yang, Ye, Li, Hou & Wang 2014, Park et al. 2014).

Without relying on the external datasets, the approaches exploit co-occurrence property between edges on depth map and ones on corresponding color image which can be classified as filter-based methods and optimization-based methods.

The pioneer work in filter-based methods is joint bilateral upsampling (JBU) framework proposed by Kopf et al. (Kopf et al. 2007). The edges of the low-resolution (LR) depth map can be refined according to the edges of the registered high-resolution (HR) color image using bilateral filtering

techniques. Liu et al. (Liu et al. 2013) propose a variant of JBU. It computes weights for average based on geodesic which is a joint space of color and distance instead of separating color space and distance space. Yang et al. (Yang, Yang, Davis & Nistér 2007) refine depth maps iteratively via cost volume which consists of a set of depth candidates. Each element in cost volume is computed by JBU with certain depth candidate. He et al. (He et al. 2010) propose a guided image filtering for depth enhancement. It models a linear relationship between the output and guiding image which is based on the assumption that the output has an edge only if the input has an edge. Min et al. (Min et al. 2012) propose a Weighted Mode Filtering method (WMF) based on joint histogram of depth candidates. It enforces the result to satisfy the requirement of L1 norm minimization which is more robust to outliers than L2 norm minimization. Lo et al. (Lo et al. 2013) propose a joint trilateral filtering (JTF) for depth enhancement. The difference between JBU and JTF is that this filter integrates local gradient information of the LR depth map. Hua et al. (Hua et al. 2016) propose a depth SR approach which adopts onion-peeling filtering procedure to exploit local gradient information of depth maps. Filter-based methods perform fast but the ability of denoising is limited due to local solution.

Compared with filter-based methods, optimization-based methods are more robust to noise. Diebel et al. (Diebel & Thrun 2005) model depth map SR as solving a multi-labeling optimization problem via MRF. Lu et al. (Lu, Min, Pahwa & Do 2011) extend this work by designing a data term which can better fits to the characteristics of depth maps. Zhu et al. (Zhu, Wang, Gao & Yang 2010) update the traditional spatial MRF to dynamic MRF which introduces both spatial and temporal information in an energy function to improve the accuracy and the robustness of depth map SR for dynamic scenes. Park et al. (Park et al. 2014) propose a non-local term to regularize depth maps and combined it with a weight scheme which involves edge, gradient, and segmentation information extracted from HR color images. Ferstl et al. (Ferstl et al. 2013) model regularization term as a sec-

ond order total generalized variation smoothness constraint and guided the depth map SR with an anisotropic diffusion tensor which is computed from the registered HR color image. Yang et al. (Yang, Ye, Li, Hou & Wang 2014) achieve depth enhancement via the color-guided auto-regression model (AR). The AR predictor for each pixel is constructed according to both the local correlation on the initial depth map and the nonlocal similarity in the registered high-quality color image. Liu et al. (Liu, Chen, Yang & Wu 2017) design the regularization term base on a robust M-estimator to implicitly handle the inconsistency between depth map and registered color image.

## 4.2 Challenges in Guided Depth Enhancement

Basically, the quality of depth edges in low resolution can be improved based on the guidance provided by the corresponding color edges. However, the assumption that the depth edges and the color edges at the corresponding locations are consistent is not always true. The incorrect guidance from the companion color image will lead to texture-copying artifacts and blurring depth edges on the enhanced depth map. Texture-copying artifacts are caused by the situation that actual smooth depth regions correspond to color regions with rich texture. By contrast, blurring depth edges are normally see in the case that smooth color regions correspond to depth regions with edges. Please refer Section 1.1 which illustrates the edge inconsistency explained above.

There have been several works (Park et al. 2014, Yang, Ye, Li, Hou & Wang 2014, Hua et al. 2016, Choi & Jung 2014) which balance the contribution from the original depth map and the companion color image to solve these challenges, but such methods have two drawbacks: 1. They do not explicitly evaluate the edge inconsistency between the color image and the corresponding depth map. Therefore, they cannot adaptively control the efforts of the guidance from the color image when enhancing the depth map. 2. The edge guidance affinities in MRF energy function are computed only

based on color and depth differences between the pixel and its neighbor pixels on color image and coarsely interpolated depth map independently. Such non-structural computing scheme ignores local depth structure.

This chapter proposes three methods to solve these challenges which provide progressive improvement. More specifically, proposed hard-decision and soft-decision edge inconsistency measurement are to solve drawback 1, and proposed Minimum Spanning Forest aims to solve drawback 2. Such proposed methods are embedded into Markov Random Field (MRF) which are explained in details in the following sections.

## 4.3 Color-Guided Depth SR via MRF Embedded with Hard-decision Edge Inconsistency Measurement

To evaluate edge inconsistency between low-resolution (LR) depth map and high-resolution (HR) color image, a few specific points should be discussed.

- To measure the inconsistency between the depth edge map and the color edge map, the resolutions of these two edge maps must be the same. In this case, the depth map in lower resolution or with holes is coarsely interpolated to the same resolution as HR color image through gridded or scattered interpolation methods before edge detection.
- Because the color image and the corresponding depth map have the structural similarity which is clearly observed on the relevant binary edge maps, the proposed method measures the inconsistency between the binary edge maps generated from the color image and the corresponding depth map respectively.

Canny operator (Canny 1986) is adopted to detect the edges of the coarsely interpolated depth map and the corresponding color image. Due to low resolution or noise, depth edges may shift from their true positions

on depth edge map. So compared with registered color edges, inconsistent depth edges can be classified into two types; the one is degraded by coarse interpolation but can be refined via guidance from high-quality color edges, the other one is edge inconsistency shown in Section 1.1. A direct solution is to classify such two cases for all edge pixels via hard-decision which definitely determines the real case. Motivated by the number of errors must be less than the capability in error correction coding, it is assumed that depth edge is degraded by coarse interpolation when the displacement between it and the nearest color edge are less than a threshold. Otherwise, it is treated as true edge inconsistency.

### 4.3.1 Energy Function Construction for MRF

According to the Hammersely-Clifford theorem (Hammersley & Clifford 1971), the proposed method defines the objective function for depth SR as Eq. (4.1).

$$\begin{aligned} \mathbf{D}^{\star} = \arg\min_{d_p \in \mathbf{D}} \sum_{o_p \in \mathbf{O}} \lambda_b^p E_{data}\left(d_p, o_p\right) + \lambda \sum_{p} \sum_{q \in \mathbf{N_p}} \lambda_s^{pq} E_{reg}\left(d_p, d_q\right) \\ E_{data}\left(d_p, o_p\right) = \left|d_p - o_p\right| \\ E_{reg}\left(d_p, d_q\right) = \left|d_p - d_q\right| \end{aligned} \tag{4.1}$$

where $\mathbf{D}$ is the depth map. $\mathbf{O}$ consists of observed depth values. $p, q$ are pixel locations on the enhanced depth map. The observed depth value of $p$ is $o_p$. $\mathbf{N_p}$ is the set of 4-connected neighboring pixels of $p$. $E_{data}$ is the data term which indicates the compatibility of enhanced depth with the observed values. $E_{reg}$ is the regularization term which leads to a piecewise smooth solution and penalizes the different depth assignments for neighboring pixels. $\lambda$ is used to balance the data term and regularization term. $\lambda_b^p$ represents confidence of observed depth value $o_p$. $\lambda_s^{pq}$ stands for the anisotropic affinity of $p, q$ which is embedded with proposed hard-decision edge inconsistency measurement. $\lambda_b^p$ and $\lambda_s^{pq}$ are explained in the following parts respectively. Such optimization problem is solved by Graph cut (Boykov et al. 2001).

### Outliers Detection

Observed depth values of the pixels located at depth edges on the low-quality depth map are unreliable, since they are blurred by mixing the depth values of two different depth layers. Therefore, they should not participate in data term construction. In the proposed method, the simple canny operator is adopted to detect the edges of low-resolution depth map. $\lambda_b^p$ is a binary value which is assigned to 0 when $p$ is an edge pixel, otherwise it is assigned to 1.

### Anisotropic Affinity Computing

For every pixel $q$ in the neighbor of pixel $p$, the anisotropic affinity $\lambda_s^{pq}$ of pixel pair $p, q$ is determined by $\lambda_s^p$ and $\lambda_s^q$ which are explained as follows:

- If $p$ consistently locates on edge or smooth region in color image and coarsely interpolated depth map, it is assumed that the color distribution are consistent with the depth distribution near $p$. $\lambda_s^p$ can be formed as weight function introduced in JBU (Kopf et al. 2007) to determine the effect of guidance from the color image which is named as "Guided by color" hereafter.

- If $p$ locates on edge in color image but not in coarsely interpolated depth map, a search window on such depth map is defined. $\lambda_s^p$ can be assigned to "Guided by color" when there are edges existing in the search window, otherwise it is classified into the situation of true edge inconsistency. Since $p$ locates in a smooth region on the depth map, $\lambda_s^p$ must be assigned to large value to suppress different label assignments for neighboring pixels, which is named as "Smooth region" for brief.

- If $p$ locates on edge in coarsely interpolated depth map but not in color image, a search window on color image is defined. $\lambda_s^p$ can be assigned to "Guided by color" when there are edges existing in the search window, otherwise, it is classified into the situation of true edge inconsistency.

Table 4.1: $\lambda_s^{pq}$ VALUE TABLE

| $\lambda_s^p$ \ $\lambda_s^q$ | Guided by color | Near edges | Smooth region |
|---|---|---|---|
| Guided by color | Guided by color | Near edges | Smooth region |
| Near edges | Near edges | Near edges | N/A |
| Smooth region | Smooth region | N/A | Smooth region |

Since $p$ locates near depth edges, $\lambda_s^p$ must be assigned to small value to encourage different label assignments for neighboring nodes, which is named as "Near edges" for brief.

According to the analysis above, $\lambda_s^p$ and $\lambda_s^q$ are assigned as Eq. (4.2).

$$\underset{\star\in\{p,q\}}{\lambda_s^\star} = \begin{cases} e^{-\frac{\nabla \mathbf{I}_{pq}^2}{\delta^2}} & \text{Guided by color} \\ e^{-\frac{\nabla \mathbf{I}_{small}^2}{\delta^2}} & \text{Near edges} \\ e^{-\frac{\nabla \mathbf{I}_{large}^2}{\delta^2}} & \text{Smooth region} \end{cases} \tag{4.2}$$

where $\nabla \mathbf{I}_{pq}$ represents the luminance difference of pixel pair $p, q$. In the proposed method, $\nabla \mathbf{I}_{small} = 1$ and $\nabla \mathbf{I}_{large} = 254$.

Because of the symmetrical relationship of $p, q$ in pixel pair, $\lambda_s^{pq}$ is determined by $\lambda_s^p$ and $\lambda_s^q$. In fact, if $\lambda_s^p$ and $\lambda_s^q$ make the same decision, $\lambda_s^{pq}$ is determined without ambiguity. In addition, according to the definition, it is noticed that if $\lambda_s^p$ is classified into "Smooth region", $\lambda_s^q$ cannot be classified into "Near edges", and vice versa. For the rest situations, if $\lambda_s^p$ is classified into "Guided by color" and $\lambda_s^q$ is classified into others, it is shown that $q$ is near the edge of either the color nor depth, therefore, $\lambda_s^{pq}$ must be determined by $\lambda_s^q$. All the situations are shown in Tab. 4.1 where the value sets of $\lambda_s^p$ and $\lambda_s^q$ are listed in the first column and row respectively. under specified choice of $\lambda_s^p$ and $\lambda_s^q$, the values of $\lambda_s^{pq}$ are shown in the table.

Table 4.2: Quantitative Upsampling Results (in MAD) on Noise-free Middlebury Datasets "Art", "Book" and "Moebius"

| Datasets / Methods | Art | | | | Book | | | | Moebius | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 2× | 4× | 8× | 16× | 2× | 4× | 8× | 16× | 2× | 4× | 8× | 16× |
| Bicubic | 0.48 | 0.97 | 1.85 | 3.59 | **0.13** | 0.29 | 0.59 | 1.15 | **0.13** | 0.30 | 0.59 | 1.13 |
| OMRF | 0.59 | 0.96 | 1.89 | 3.78 | 0.21 | 0.33 | 0.61 | 1.20 | 0.24 | 0.36 | 0.65 | 1.25 |
| JBUV | 0.55 | 0.68 | 1.44 | 3.52 | 0.29 | 0.44 | 0.62 | 1.45 | 0.38 | 0.46 | 0.67 | 1.10 |
| JBU | 0.45 | 0.85 | 1.68 | 3.35 | 0.17 | 0.36 | 0.74 | 1.56 | 0.18 | 0.37 | 0.76 | 1.46 |
| Guided | 0.63 | 1.01 | 1.70 | 3.46 | 0.22 | 0.35 | 0.58 | 1.14 | 0.23 | 0.37 | 0.59 | 1.16 |
| TGV | 0.45 | 0.65 | 1.17 | 2.30 | 0.18 | 0.27 | **0.42** | **0.82** | 0.18 | 0.29 | **0.49** | **0.90** |
| MLS | **0.27** | 0.68 | 1.04 | **2.20** | 0.16 | **0.26** | 0.48 | 1.16 | 0.15 | **0.25** | **0.49** | 0.93 |
| Pro-Hard | 0.40 | **0.56** | **1.03** | 2.38 | 0.14 | 0.27 | 0.48 | 0.92 | 0.15 | 0.30 | 0.62 | 1.20 |

### 4.3.2 Experimental Results

To evaluate performance of the proposed method, the Middlebury stereo dataset are used (*Middlebury Datasets [Online]* n.d.), which provides RGB-D image pairs. The HR depth maps are downsampled by four factors (i.e., 2×, 4×, 8× and 16×) to generate LR input. The proposed method (Pro-Hard) is compared with Bicubic interpolation (bicubic), Original MRF-based method (OMRF) (Diebel & Thrun 2005), Joint bilateral upsampling (JBU) (Kopf et al. 2007), Spatial-depth super resolution for range images (JBUV) (Yang et al. 2007), Guided image filtering (Guided) (He et al. 2010), Total generalized variation (TGV) (Ferstl et al. 2013) and moving least squares filter (MLS) (Bose & Ahuja 2006).

The objective results evaluated by MAD (mean of absolute differences) are shown in Tab. 4.2 and Tab. 4.3. Overall, the proposed method can provide satisfied results for small upsampling factor. However, the performance is not robust for large upsampling factor. As the upsampling factor increases, the quality of edge maps is degraded significantly. So the drawback of hard-decision edge inconsistency measurement is shown. Compare with the proposed method, MLS (Bose & Ahuja 2006)and TGV (Ferstl et al. 2013) perform better in the cases of large upsampling factors (e.g., 8× and 16 ×).

Table 4.3: Quantitative Upsampling Results (in MAD) on Noise-free Middlebury Datasets "Reindeer", "Laundry" and "Dolls"

| Datasets / Methods | Reindeer | | | | Laundry | | | | Dolls | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 2× | 4× | 8× | 16× | 2× | 4× | 8× | 16× | 2× | 4× | 8× | 16× |
| Bicubic | 0.30 | 0.55 | 0.99 | 1.88 | 0.28 | 0.54 | 1.04 | 1.95 | 0.20 | 0.36 | 0.66 | 1.18 |
| JBU | 0.27 | 0.50 | 1.00 | 1.89 | 0.26 | 0.49 | 0.94 | 1.95 | 0.20 | 0.38 | 0.74 | 1.46 |
| Guided | 0.42 | 0.53 | 0.88 | 1.80 | 0.38 | 0.52 | 0.95 | 1.90 | 0.28 | 0.35 | **0.56** | 1.13 |
| TGV | 0.32 | 0.49 | 1.03 | 3.05 | 0.31 | 0.55 | 1.22 | 3.37 | 0.21 | **0.33** | 0.70 | 2.20 |
| MLS | 0.32 | 0.64 | **0.74** | **1.43** | 0.23 | **0.39** | **0.81** | **1.53** | 0.24 | 0.36 | 0.61 | **0.98** |
| Pro-Hard | **0.21** | **0.40** | **0.74** | 1.50 | **0.21** | 0.47 | 0.90 | 2.02 | **0.18** | 0.37 | 0.72 | 1.44 |

### 4.3.3 Conclusion

This section proposes a depth SR method with embedded hard-decision edge inconsistency measurement. Although it provides satisfied results for small upsampling factors (i.e., 2× and 4×), the performance is not robust for large upsampling factors. The reason is that the hard-decision edge inconsistency measurement cannot work well when the quality of coarsely interpolated depth map is very low.

## 4.4 Color-Guided Depth Enhancement via MRF Embedded with Soft-decision Edge Inconsistency Measurement

Due to the drawback of hard-decision edge inconsistency measurement, this section proposes a soft-decision edge inconsistency measurement method which more precisely measures the degree of inconsistency occurring between depth edges and the corresponding color edges in a numerical way. Such edge inconsistency measurement is embedded into MRF which has stronger ability in mitigating texture-copying artifacts and preserving depth edges than hard-decision counterpart.

### 4.4.1 Modifications in MRF Energy Function

Since multi-labels optimization via graph cut cannot reach the global minimum exactly and depth value is always recorded in mm unit as a continuous floating value, e.g. ToF-Mark datasets (Ferstl et al. 2013), this section constructs MRF with continuous variables. To simplify the optimization, the energy function is modified as Eq. (4.3).

$$\begin{aligned} \mathbf{D}^{\star} = \arg\min_{d_p \in \mathbf{D}} \sum_{o_p \in \mathbf{O}} E_{data}\left(d_p, o_p\right) + \lambda \sum_{p} \sum_{q \in \mathbf{N_p}} \lambda_s^{pq} E_{reg}\left(d_p, d_q\right) \\ E_{data}\left(d_p, o_p\right) = \left(d_p - o_p\right)^2 \\ E_{reg}\left(d_p, d_q\right) = \left(d_p - d_q\right)^2 \end{aligned} \tag{4.3}$$

where $\lambda_s^{pq}$ is computed based on proposed soft-decision edge inconsistency measurement.

The main differences between this model and the one introduced in the previous section are in two aspects; 1. By assuming Gaussian noise in raw LR depth map, data term and regularization term are defined in square function instead of absolute counterpart for better denoising performance. 2. Due to the drawback of hard-decision edge inconsistency measurement, the affinities of regularization term in this section are computed by soft-decision edge inconsistency measurement which is expressed in the following subsections.

### 4.4.2 Soft-decision Edge Inconsistency Measurement

Motivated by (Jang & Kim 2012), the inconsistency measurement between the color edge map and the depth edge map can be modeled as a bi-directional edge map quality assessment. However, in (Jang & Kim 2012), common edge map quality measurement is based on the position shift of each edge pixel against the position on the ground truth. The case to be investigated here is different. In the case of depth enhancement, the matched edge pixels on the depth edge map and the color edge map which should have located in the same position always have displacement with each other. The reasons are some pre-processing such as coarse interpolation as explained above or

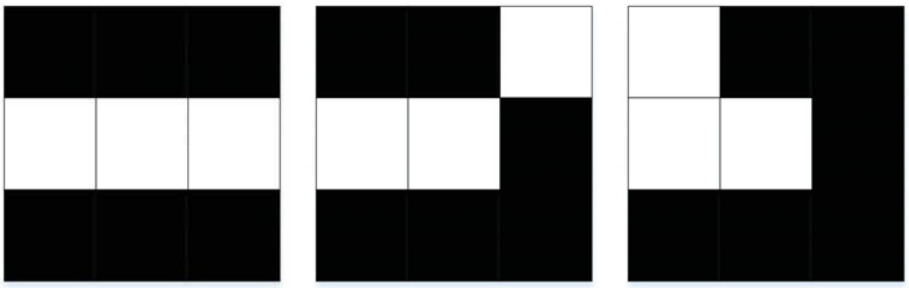
noise in depth sensors. Thus, it is impossible to measure the inconsistency on the difference between the positions of each pair of matched edge pixels like the existing edge quality measurement methods (Jang & Kim 2012, Prieto & Allen 2003). Instead, the proposed edge inconsistency measurement method is based on the structure similarity of the edge maps which considers the structure presented by local neighboring regions as well as the global structure of the whole edge map. For convenience, the following explanation is based on the reference edge map and the target edge map whose meaning can be found in the end of this part.

For each edge pixel on the reference edge map, it will search the best consistency on the target edge map within a neighboring region around the corresponding position. This implies that if the color edge and the depth edge are consistent, the displacement of matched edge pixels should be constrained in a small range. Moreover, strength and orientation of the displacements of all matched edge pixels in a nearby region should be consistent. These two constraints are solved in an MRF optimization through its data term and regularization term respectively as Eq. (4.4). The data term implies local structure information and regularization term implies global structure information. Therefore, the edge inconsistency measurement is robust to the errors in the original depth edge map. It should be addressed that this MRF problem is to perform edge inconsistency measurement but not for the depth enhancement task.

$$\mathbf{L}^{\star} = \arg\min_{l \in \mathbf{L}} \sum_{p \in \mathbf{ref}} C\left(p, p + l_p\right) + \mu \sum_{p \in \mathbf{ref}} \sum_{q \in \mathbf{N_p}} V\left(l_p, l_q\right) \tag{4.4}$$

where $C$ and $V$ are functions defined for data term and regularization term in MRF respectively. $\mu$ is a balance factor between data term and regularization term. It is set to 0.1 in the proposed method. $p$ represents the position of an edge pixel on the reference edge map **ref**. $\mathbf{N_p}$ is the set of 8-connected neighboring pixels of $p$. $l_p$ which is an element of $\mathbf{L}$ stands for the displacement for $p$. Therefore, $p+l_p$ represents the position of the edge pixel $k$ on the target edge map. Since the sub-pixels created virtually by interpolation pro-

cess may not be stable, each edge pixel in the reference edge map is mapped to the existing edge pixel detected in the target edge map. In other words, the proposed method assigns $\mathbf{L}$ in integer pixel precision and this is a discrete MRF optimization problem. In all experiments, the size of the searching window determines the range of $l_p$ which is 5×5 for 2× SR, 7×7 for 4× SR, 9×9 for 8× SR, 11×11 for 16× SR and 7×7 for depth map completion without SR. The data term $C(p, k)$ is the cost of matching the reference edge pixel $p$ against the target edge pixel $k$. Given $p$, if corresponding target pixel $k$ on the target edge map is not an edge pixel, it is regarded as definite inconsistency under current $l_p$. In that case, $C(p, k)$ is assigned to the maximum inconsistency value (i.e., 1 in the proposed method). Otherwise, this inconsistency is measured on two patches where the edge pixel $p$ and the edge pixel $k$ are the center positions respectively. In the proposed method, the size of the patch is 3×3. This measurement is sorted out through Minimum Weighted Bipartite Matching (Kuhn 1955) which is more robust than Mean of Absolute Difference (MAD). Actually, Minimum Weighted Bipartite Matching is based on structure similarity. However, MAD only considers difference of each pixel pair independently. In a weighted bipartite graph, each graph edge has an associated value. A Minimum Weighted Bipartite Matching is to find the best matching where the sum of the values of graph edges (graph edges are the set of arcs or lines in graph theory) linking matched vertices is a minimum. In the propose method, the quality of the bipartite matching is measured according to the difference between the locations of the matched edge pixels and the amounts of edge pixels in two patches. Fig. 4.1 provides an illustration on the advantage of Weighted Bipartite Matching compared with MAD. In the three patches shown in Fig. 4.1, white pixels and black pixels represent edge pixels and non-edge pixels respectively. When MAD is applied, both (b) and (c) (i.e., the target patches) have the same matching cost to the reference patch a). However, it can be observed that, in term of local structure, target patch b) is closer to a). Such fine-grained level similarity can be successfully picked up by Bipartite Graph Matching used in



**(a)** Reference patch **(b)** Target patch 1 **(c)** Target patch 2

**Figure 4.1:** An illustration on the advantage of Weighted Bipartite Matching

the proposed method. Based on the analysis above, the data term $C(p,k)$ is expressed as Eq. (4.5).

$$C(p,k) = \begin{cases} 1\,(definitely\ inconsistency), & \text{if } k \notin edge\ pixels \\ BM\,(\mathbf{R_p}, \mathbf{R_k}, \mathbf{E}, \mathbf{W}), & \text{otherwise} \end{cases} \tag{4.5}$$

where $BM$ stands for Minimum Weighted Bipartite Matching (Kuhn 1955). The bipartite graph $\mathbf{G}(\mathbf{R_p}, \mathbf{R_k}, \mathbf{E}, \mathbf{W})$ is defined as follows; $\mathbf{R_p}$ and $\mathbf{R_k}$ are vertices. $\mathbf{E}$ represents graph edges between vertices and $\mathbf{W}$ is the vector which assigns weight to each graph edge in $\mathbf{E}$. Specifically, $\mathbf{R_p} = \{ep_1, ep_2, \ldots ep_m\}$ and $\mathbf{R_k} = \{ek_1, ek_2, \ldots ek_m\}$ represent the sets of edge pixels in the two patches (excluding $p$ and $k$ which are the center edge pixels of these two patches). $m$ and $n$ are the amount of edge pixels inside these two sets respectively. Thus, the inconsistency measurement between $p$ and $k$ is regarded as a matching problem between two data sets $\mathbf{R_p}$ and $\mathbf{R_k}$. In addition, the locations of an edge pixel and its true matched edge pixel are assumed to be close to each other. This assumption complies with the similarity of local structural information. Therefore, each element of $\mathbf{W}$ is defined as $\phi(ep_i, ek_j)$ which is a monotonic function that returns a positive

penalty for local structural matching.

$$\phi\left(ep_i, ek_j\right) = f\left(\left|ep_i^x - ek_j^x\right| + \left|ep_i^y - ek_j^y\right|\right) \tag{4.6}$$

where $f\left(0\right) = 0$, $f\left(1\right) = 1$, $f\left(2\right) = 1.6$ and $f\left(x\right) = 2,\ if\ x > 2$. $ep_i^x$ and $ep_i^y$ are the coordinates of the edge pixel $ep_i$ respectively.

Minimum Weighted Bipartite Matching (Kuhn 1955) is employed to enforce one-to-one matching between the edge pixel data sets above. That is, it assures any edge pixel in $\mathbf{R_p}/\mathbf{R_k}$ matches at most one edge pixel in $\mathbf{R_k}/\mathbf{R_p}$ with $|m-n|$ unmatched pixels. Fig. 4.2 gives an illustration of Minimum Weighted Bipartite Matching with unmatched pixels marked. Actually, the amount of unmatched pixels also reflects the structure differences between the edge pixel sets $\mathbf{R_p}$ and $\mathbf{R_k}$. Furthermore, to effectively mitigate the effect of edge detection errors in noisy depth maps, the difference reflected by these unmatched pixels should be taken into account. It can be observed that when the numbers of edge pixels in both patches are very different, the proposed method considers that this edge may be caused by noise or this matching is not reliable. Therefore, it should add large cost to this matching. To consider these issues above, the edge inconsistency measurement term $BM\left(\mathbf{R_p}, \mathbf{R_k}, \mathbf{E}, \mathbf{W}\right)$ in Eq. (4.5) is carefully adjusted and defined as Eq. (4.7)

$$\begin{aligned} &BM\left(\mathbf{R_p}, \mathbf{R_k}, \mathbf{E}, \mathbf{W}\right) = \\ &\left(\sum_{(mp_s, mk_s) \in \mathbf{R'_{pk}}} \phi\left(mp_s, mk_s\right)/2 + |m-n|\right)/8 \end{aligned} \tag{4.7}$$

where $\mathbf{R'_{pk}} = \left\{\left(mp_1, mk_1\right), \left(mp_2, mk_2\right), \ldots \left(mp_r, mk_r\right)\right\}$ is the set of edge pixel pairs selected by Minimum Weighted Bipartite Matching (Kuhn 1955). $\phi\left(mp_s, mk_s\right)$ is the weight of the edge linking the edge pixels $mp_s$ and $mk_s$ with $s = \left\{1, 2 \ldots r\right\}$. Therefore, $\sum_{(mp_s, mk_s) \in \mathbf{R'_{pk}}} \phi\left(mp_s, mk_s\right)$ is the matching cost of Minimum Weighted Bipartite Matching mentioned above. In order

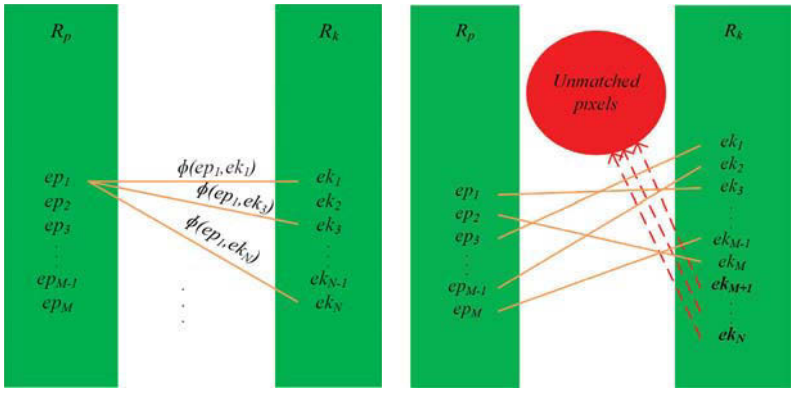



**Figure 4.2:** An illustration of Minimum Weighted Bipartite Matching problem with (a) configuration of a weighted bipartite graph and (b) a result of Minimum Weighted Bipartite Matching with unmatched pixels marked in bold.

to constrain the data term $C(p,k)$ in the range of [0, 1], the common normalization process is applied on Eq. (4.7).

$V(l_p, l_q)$ is the regularization term in Eq. (4.4), which gives a penalty when adjacent edge pixels have different displacements as,

$$\underset{q\in\mathbf{N_p}}{V(l_p,l_q)} = \begin{cases} 0, & if\ l_p = l_q \\ 1, & otherwise \end{cases} \tag{4.8}$$

where $l_p$ is the same as Eq. (4.4), representing the displacement vector for the edge pixel $p$.

Once the data term and the regularization term in Eq. (4.4) are defined (see Eq. (4.5) and Eq. (4.8)), Graph cut (Boykov et al. 2001) is adopted to solve the multiple label MRF optimization problem. Then, the inconsistency measurement for edge pixel $p$ represented by $C(p,k)$ can be computed by the optimized displacement $l_p^\star \in \mathbf{L}^\star$. In this proposed method, the inconsistency map $\mathbf{C_{ref}}$ is the set of $C\left(p, p+l_p^\star\right)$, representing the inconsistency measurements for all edge pixels in the reference edge map. If there is no matching found for certain edge pixel, the displacement $l$ of this edge pixel is

meaningless. And its inconsistency value is assigned to maximum value (i.e., 1 in this work).

The edge inconsistency is measured based on the reference edge map against the target edge map. Thus, the measurement results will be different when swapping these two edge maps. In this work, the two edge maps are the color edge map and the corresponding depth edge map. When the color edge map is regarded as the reference edge map, it can be observed that the most inconsistent positions detected reflect the texture-copying happening areas. On the other hand, when the depth edge map is regarded as the reference edge map, it is observed that the most inconsistent positions detected reflect happening areas of blurring depth edges. Fig. 4.3 illustrates the bi-direction inconsistency measurement for Middlebury dataset Art which is expressed in false color images. In Fig. 4.3c and Fig. 4.3d, the color along the edge pixels represents the strength of inconsistency of the edges between the reference edge map and the target edge map. According to the color scale coding in Fig. 4.3e, the color code on the leftmost side (i.e., dark blue) means the most consistent case. On the contrary, the color code on the rightmost side (i.e., dark red) means the most inconsistent case.

After the bi-direction evaluation, there are two inconsistency maps $\mathbf{C_{color}}$ (the color edge map is regarded as the reference edge map), $\mathbf{C_{depth}}$ (the depth edge map is regarded as the reference edge map) as well as two sets of displacements $\mathbf{L_{color}}$, $\mathbf{L_{depth}}$ (defined in the same way as $\mathbf{C_{color}}$ and $\mathbf{C_{depth}}$) for an image pair. Before embedding the inconsistency measurement values into the proposed MRF-based model, these two inconsistency maps must be consolidated with each other.

As mentioned before, the positions of edge pixels on the coarsely interpolated depth map are unreliable. On the contrary, the positions of edge pixels on the color edge map are more precise because of high quality of the color image. Through the solution of the MRF optimization problem in Eq. (4.4) with the depth edge map as the reference edge map, the displacement between each depth edge pixel $p$ and its matched color edge pixel $k$ is

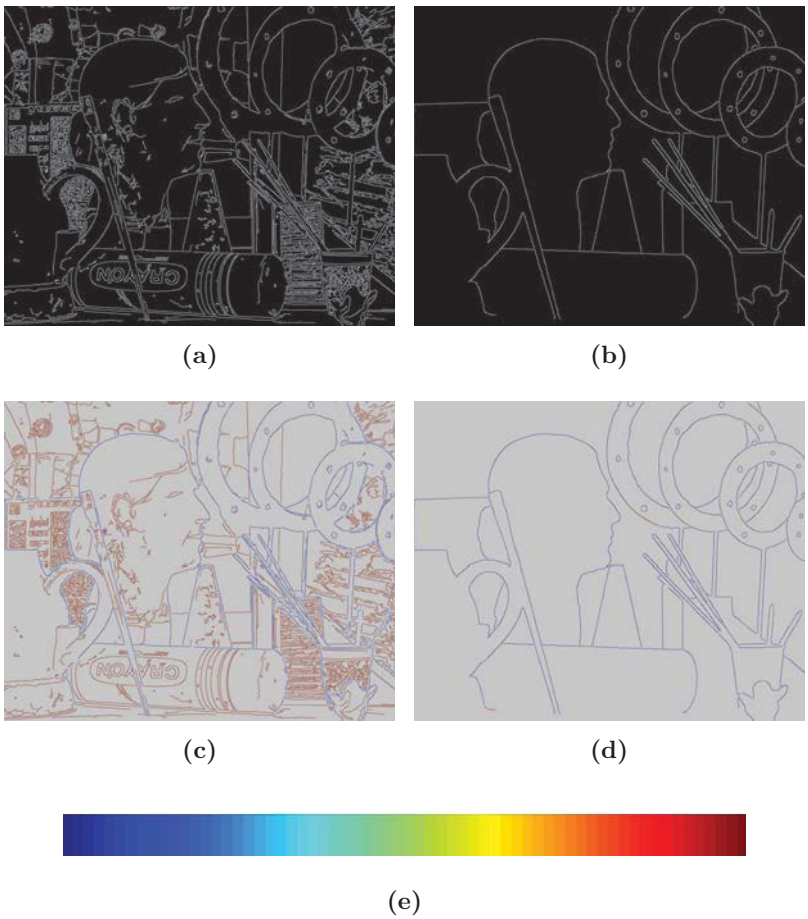


**Figure 4.3:** The bi-direction inconsistency measurement for Middlebury dataset "Art, (a) the color edge map, (b) the depth edge map, (c) the edge inconsistency measurement in the case that the color edge map is the reference edge map, (d) the inconsistency measurement in the case that the depth edge map is the reference edge map, (e) the color scale coding (The color code on the leftmost side (i.e., dark blue) means the most consistent case between two edge maps. The color code on the rightmost side (i.e., dark red) means the most inconsistent case between two edge maps). The inconsistency values of non-edge pixels in (b) and (d) are unavailable. They are shown in gray which is out of the color scale coding (e).

$\mathbf{L_{depth}}(p)$. Consequently, the true location of the observed depth edge pixel $p$ supposes to be more close to $p + \mathbf{L_{depth}}(p)$ when $\mathbf{C_{depth}}(p) \neq 1$. For the case of definite inconsistency $\mathbf{C_{depth}}(p) = 1$, the position of the edge pixel $p$ is unchanged because there is no matched edge pixel in the color edge map. Moreover, due to the uncertainty of displacements, given a color edge pixel $p'$, it may correspond to more than one depth edge pixels $p$. Under this situation, the inconsistency value of the best mapping edge pixel position with the lowest cost is updated. In the proposed method, the adjusted $\mathbf{C'_{depth}}$ is expressed as,

$$\begin{aligned} \mathbf{C'_{depth}}(p') &= \min_{p \in \left\{p | p' = p + \mathbf{L_{depth}}(p)\right\}} \mathbf{C_{depth}}(p), \ if \ \mathbf{C_{depth}}(p) \neq 1 \\ \mathbf{C'_{depth}}(p) &= \mathbf{C_{depth}}(p), \qquad otherwise \end{aligned} \tag{4.9}$$

Once two inconsistency maps $\mathbf{C'_{depth}}$ and $\mathbf{C_{color}}$ are aligned, a confidence map $\boldsymbol{\alpha}$ is defined as Eq. (4.10) which has considered the inconsistency measurement in the bi-direction calculation. It describes the final inconsistency status between the color edge map and the depth edge map.

$$\boldsymbol{\alpha} = \max\left(\mathbf{C'_{depth}}, \mathbf{C_{color}}\right) \tag{4.10}$$

In the next subsection, in order to improve the performance of color-guided depth enhancement, this measurement is applied into MRF to fine tune the efforts of the guidance from the color image.

### 4.4.3 Embedding Edge Inconsistency Measurement into MRF

Generally speaking, guidance information for depth enhancement task can be derived from two sources. One is from the registered color image, and the other is from the original depth map itself. Based on the confidence map $\boldsymbol{\alpha}$ computed in Eq. (4.10), the proposed method combines these two kinds

of information systematically to generate a new guidance for computing the anisotropy affinity $\lambda_s^{pq}$.

$$\lambda_s^{pq} = e^{\frac{-\left(|\nabla_c^{pq}|(1-\alpha_{pq})+\left|\nabla_d^{pq}\right|\alpha_{pq}\right)^2}{2\delta^2}} \tag{4.11}$$

where $\nabla_c^{pq}$ and $\nabla_d^{pq}$ represent color difference and depth difference between the position $p$ and its neighboring pixel $q$ in the guided color image and the coarsely interpolated depth map respectively. $\delta$ controls bandwidth of the exponential function. In addition, as mentioned above, only edge pixels have available confidence values. To compute the robust confidence value for each pixel pair $p, q$, max operation is adopted to select the more inconsistent one between the confidence values of $p$ and $q$ ($\boldsymbol{\alpha}(p)$, $\boldsymbol{\alpha}(q)$). It can mitigate texture-copying artifacts and preserve depth edges better because of the single pixel width edges detected by Canny operator. $\alpha_{pq}$ is expressed as $\alpha_{pq} = \max(\boldsymbol{\alpha}(p), \boldsymbol{\alpha}(q))$. More specifically, when neighboring pixel pair $p, q$ are located across the edges in the color image as well as the depth map, $\alpha_{pq}$ is more close to 0 and $\nabla_c^{pq}$ plays a more important role in computing the anisotropy affinity $\lambda_s^{pq}$. In such situation, the guidance from the registered color image helps recover sharp edges in the reconstructed depth map. By contrast, when neighboring pixel pair $p, q$ only across the edge either on the color image or on the depth map, but not both, $\alpha_{pq}$ is more close to 1 and $\nabla_d^{pq}$ provides main guidance. In these two situations, depth enhancement is through the approach of single depth map enhancement method. Indeed, some single depth enhancement method can be adopted to provide more accuracy depth map instead of simple interpolated depth map. However, the improvement is not significant when upsampling factor is small. On the other hand, it is difficult to obtain accurate depth edges for large upsampling factor by using single depth enhancement methods. Therefore, by considering the complexity and equity, the proposed method uses coarse interpolated depth map as the guidance source for all cases. The benefit of using $\nabla_d^{pq}$ is twofold. On the one hand, it mitigates texture-copying artifacts. On the other hand, the guidance from the interpolated depth map is more reasonable than the

incorrect guidance from the color image.

The scenario discussed above is on the regions around edge pixels. For pixels located on smooth regions where there is no edge pixel on neither the color image nor the coarsely interpolated depth map, Eq. (4.11) cannot satisfy such case because it is impossible to calculate the edge inconsistency in a local region where there is no edge pixel at all. In the proposed method, it is updated as Eq. (4.12) for this special case, where the guidance information for depth enhancement is from the coarsely interpolated depth map only to better mitigate texture-copying artifacts.

$$\lambda_s^{pq} = e^{\frac{-\left(\nabla_d^{pq}\right)^2}{2\delta^2}} \tag{4.12}$$

Based on the analysis above, the proposed method can preserve depth edges and mitigate texture-copying artifacts efficiently by adaptively controlling the efforts of the guidance from the color image.

In addition, in regions near depth edges, $\delta$ should be small to preserve depth edges. By contrast, $\delta$ should be large to suppress noise in smooth regions. The proposed method assigns different values to $\delta$ for smooth regions and non-smooth regions respectively which are determined by the depth edge map. More specifically, on the depth edge map, if there is no edge pixel in the local windows centered at $p$ and its neighboring pixel $q$ respectively, the pixel pair $p, q$ is located at a smooth region. Otherwise, such pixel pair is located at a non-smooth region. In this method, $\delta$ is set to 2 and 4 for non-smooth regions (Eq. (4.11)) and smooth regions (Eq. (4.12)) respectively.

### 4.4.4 Algorithm Complexity Discussion

In the edge inconsistency measurement stage, the multi-label Graph Cut problem is solved by several binary-label Graph Cut sub-problems through $\alpha - expansion$ method. The complexity of binary-label Graph Cut is up to $O\left(MN^2\left|C\right|\right)$, where $M$ and $N$ are the number of graph edges and node (i.e., the number of edge pixels detected in the reference edge map) in the graph

Table 4.4: Average Running Time of the Proposed Method (16×)

| Running time on average | Bi-inconsistency measurement | MRF Optimization | Total |
|---|---|---|---|
| Unit:Second | 69.72 | 45.67 | 115.39 |

respectively. And $|C|$ is the cost of the minimum cut which is the smallest total weight of the edges which if removed would disconnect the source from the sink (Boykov & Kolmogorov 2004). Therefore, the complexity of multi-label Graph Cut is up to $O(LMN^2|C|)$, where $L$ is the number of labels (Boykov et al. 2001). In addition, the complexity of the Hungarian algorithm (Kuhn 1955) for Minimum Weighted Bipartite Graph Matching is $O(V^2E)$, where $V$ and $E$ represents the number of vertices (i.e., the number of edge pixels in the two patches) and graph edges respectively.

### 4.4.5 Experimental Results

The platform to carry out the experiments is a PC with Intel i7 2.60 GHz, 12G RAM. The plain Matlab implementation of the proposed method (Graph Cut is implemented in C code) takes 115.39s on average to upsample the low quality depth map up to the resolution of 1376×1088 in the case of 16×. The running time of each step is listed in Tab. 4.4.

Experiments consist of three parts. The first part is to evaluate the proposed methods performance on Middlebury datasets (*Middlebury Datasets [Online]* n.d.) in which the synthetic depth maps are degraded manually in various ways. The comparison performance between the proposed method and several existing methods are shown. The second part is to apply our method on real datasets (ToF-Mark datasets (Ferstl et al. 2013) and NYU datasets (Silberman, Hoiem, Kohli & Fergus 2012)) to obtain high quality depth maps in order to show the robustness of the proposed method in real scenes. The third part is to demonstrate the performance of the proposed method on depth map enhancement which is to tackle a difficult situation

when the complex degradation occurs. It involves both lower resolution and significant holes.

Regarding $\lambda$ in Eq. (4.3), it can be theoretically analysed through two aspects which are based on the upsampling factor and the noise situation on the LR depth map. On the one hand, $\lambda$ should decrease as the upsampling factor increases. A larger upsampling factor will cause sparsity in the pixel set which have observed depth values, so the contribution of the data term in MRF is light. To balance the contributions of the data term and the regularization term, it is necessary to reduce $\lambda$ thus the contribution of the data term will be lift up relatively even the observed depth data is sparse. On the other hand, increasing $\lambda$ for the case of stronger noise is able to provide the MRF model more robustness to noise by enhancing the efforts of the regularization term. In this work, it is also observed that when depth map downsampled by nearest neighbor interpolation is noise-free, the upsampling factor has less impact to $\lambda$. That is, for different upsampling factors, the optimal $\lambda$ may have the close values as long as noise on the depth map is not significant. Through cross-validation process, the proposed method fixes $\lambda$ to 0.01 for all upsampling experiments in which the LR depth maps are from Middlebury dataset without adding noise. For the case of adding noise, the relation between $\lambda$ and the upsampling factor is defined as,

$$\lambda = \frac{\kappa}{factor}, \ if \ factor > 1 \tag{4.13}$$

where $\kappa$ is a constant, it is set to 3.2 in all depth map SR experiments. $factor$ is the upsampling factor. In addition, $\lambda$ is fixed to 5 for all the depth map completion experiments.

### Experiments on Datasets with Synthetic Degradations

In this subsection, six datasets including "Art", "Book", "Moebius", "Reindeer", "Laundry", and "Dolls" from the Middleburys benchmark (*Middlebury Datasets [Online]* n.d.) are used for the evaluation. Three kinds of degradations are considered in experiments which are 1) downsampling, 2) down-

sampling with adding noise and 3) structural error and random missing.

1) Degradation by down-sampling

The proposed method (Pro-Soft) is compared with 12 benchmark and the state-of-the-art methods: Bicubic interpolation (bicubic), Joint bilateral upsampling (JBU) (Kopf et al. 2007), Improved JBU with $\lambda_s^{pq}$ (IMJBU), Original MRF-based method (OMRF) (Diebel & Thrun 2005), Spatial-depth super-resolution for range images (JBUV) (Yang et al. 2007), Joint geodesic filtering (JGF) (Liu et al. 2013), Guided image filtering (Guided) (He et al. 2010), Total generalized variation (TGV) (Ferstl et al. 2013), Edge-weighted NLM-regularization (NLMR) (Park et al. 2014), Moving least squares filter (MLS) (Bose & Ahuja 2006), Auto-regression model (AR) (Yang, Ye, Li, Hou & Wang 2014) and the method proposed in the previous section (Pro-Hard). Moreover, it is realized that the existing papers OMRF (Diebel & Thrun 2005) and JBUV (Yang et al. 2007) did not report the experimental results on the datasets of "Reindeer, "Laundry and "Dolls.

Tab. 4.5 and Tab. 4.6 show the upsampling results under four different upsampling factors with optimal and suboptimal results marked in bold and underlined respectively. It is noticed that the proposed method obtains the lowest MAD for most cases. In the case of 16× SR, the coarsely upsampled depth map introduces significant errors, which affects the quality of the depth edge map. However, the performance of the proposed method in such case achieves the best ones in 2 out of 6 cases, sub-optimal ones in 2 out of 6 cases. In the rest cases, the proposed method achieves the performance on top rank 3. It is shown that the proposed method is robust to the quality of the depth edge map. In addition, improved JBU (IMJBU) can improve the performances of JBU a little. However, the overall performances are worse than global optimization-based methods, such as AR (Yang, Ye, Li, Hou & Wang 2014) and TGV (Ferstl et al. 2013).

Fig. 4.4 shows the experimental results of 8× upsampled depth maps (where the specific details can be seen by zooming in the image) for "Dolls" dataset compared with 5 state-of-the-art methods: NLMR (Park et al. 2014),

Table 4.5: Quantitative Upsampling Results (in MAD) on Noise-free Middlebury Datasets "Art", "Book" and "Moebius"

| Datasets / Methods | Art | | | | Book | | | | Moebius | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 2× | 4× | 8× | 16× | 2× | 4× | 8× | 16× | 2× | 4× | 8× | 16× |
| Bicubic | 0.48 | 0.97 | 1.85 | 3.59 | 0.13 | 0.29 | 0.59 | 1.15 | 0.13 | 0.30 | 0.59 | 1.13 |
| OMRF | 0.59 | 0.96 | 1.89 | 3.78 | 0.21 | 0.33 | 0.61 | 1.20 | 0.24 | 0.36 | 0.65 | 1.25 |
| JBUV | 0.55 | 0.68 | 1.44 | 3.52 | 0.29 | 0.44 | 0.62 | 1.45 | 0.38 | 0.46 | 0.67 | 1.10 |
| JBU | 0.45 | 0.85 | 1.68 | 3.35 | 0.17 | 0.36 | 0.74 | 1.56 | 0.18 | 0.37 | 0.76 | 1.46 |
| IMJBU | 0.43 | 0.83 | 1.62 | 3.26 | 0.16 | 0.34 | 0.72 | 1.47 | 0.17 | 0.36 | 0.74 | 1.39 |
| Guided | 0.63 | 1.01 | 1.70 | 3.46 | 0.22 | 0.35 | 0.58 | 1.14 | 0.23 | 0.37 | 0.59 | 1.16 |
| NLMR | 0.41 | 0.65 | 1.03 | 2.11 | 0.17 | 0.30 | 0.56 | 1.03 | 0.18 | 0.29 | 0.51 | 1.10 |
| JGF | 0.29 | 0.47 | 0.78 | 1.54 | 0.15 | 0.24 | 0.43 | 0.81 | 0.15 | 0.25 | 0.46 | 0.80 |
| TGV | 0.45 | 0.65 | 1.17 | 2.30 | 0.18 | 0.27 | 0.42 | 0.82 | 0.18 | 0.29 | 0.49 | 0.90 |
| MLS | 0.27 | 0.68 | 1.04 | 2.20 | 0.16 | 0.26 | 0.48 | 1.16 | 0.15 | 0.25 | 0.49 | 0.93 |
| AR | **0.18** | 0.49 | **0.64** | 2.01 | 0.12 | 0.22 | **0.37** | 0.77 | **0.10** | **0.20** | 0.40 | **0.79** |
| Pro-Hard | 0.40 | 0.56 | 1.03 | 2.38 | 0.14 | 0.27 | 0.48 | 0.92 | 0.15 | 0.30 | 0.62 | 1.20 |
| Pro-Soft | **0.18** | **0.45** | 0.71 | **1.97** | **0.10** | **0.20** | **0.37** | **0.74** | **0.10** | **0.20** | **0.39** | 0.80 |

MLS (Bose & Ahuja 2006), JGF (Liu et al. 2013), AR (Yang, Ye, Li, Hou & Wang 2014) and TGV (Ferstl et al. 2013). From the highlighted regions, it is shown that NLMR, MLS, JGF and TGV severely suffer from blurring depth edges and texture-copying artifacts. AR provides comparable results to that of the proposed method, but it does not well deal with texture-copying artifacts and blurring depth edges either. Compared with the methods above, the proposed method generates the best depth map SR results.

2) Degradation by down-sampling with adding noise

In real situation, depth maps captured by sensors are accompanied by unavoidable noise. To simulate such cases, tests are run on the datasets provided by AR (Yang, Ye, Li, Hou & Wang 2014) which introduces Gaussian noise with zero mean, variance of 25 to the downsampled datasets at four upsampling factors. Tab. 4.7 and Tab. 4.8 give the depth enhancement results of the proposed method as well as 7 benchmark and the state-of-the-art methods with optimal and suboptimal results marked in bold and underlined

Table 4.6: Quantitative Upsampling Results (in MAD) on Noise-free Middlebury Datasets "Reindeer", "Laundry" and "Dolls"

| Datasets / Methods | Reindeer | | | | Laundry | | | | Dolls | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 2× | 4× | 8× | 16× | 2× | 4× | 8× | 16× | 2× | 4× | 8× | 16× |
| Bicubic | 0.30 | 0.55 | 0.99 | 1.88 | 0.28 | 0.54 | 1.04 | 1.95 | 0.20 | 0.36 | 0.66 | 1.18 |
| JBU | 0.27 | 0.50 | 1.00 | 1.89 | 0.26 | 0.49 | 0.94 | 1.95 | 0.20 | 0.38 | 0.74 | 1.46 |
| IMJBU | 0.27 | 0.49 | 0.98 | 1.87 | 0.25 | 0.48 | 0.92 | 1.94 | 0.20 | 0.37 | 0.73 | 1.44 |
| Guided | 0.42 | 0.53 | 0.88 | 1.80 | 0.38 | 0.52 | 0.95 | 1.90 | 0.28 | 0.35 | 0.56 | 1.13 |
| NLMR | 0.20 | 0.37 | 0.63 | 1.28 | 0.17 | 0.32 | 0.54 | 1.14 | 0.16 | 0.31 | 0.56 | 1.05 |
| JGF | 0.23 | 0.38 | 0.64 | 1.09 | 0.21 | 0.36 | 0.64 | 1.20 | 0.19 | 0.33 | 0.59 | 1.06 |
| TGV | 0.32 | 0.49 | 1.03 | 3.05 | 0.31 | 0.55 | 1.22 | 3.37 | 0.21 | 0.33 | 0.70 | 2.20 |
| MLS | 0.32 | 0.64 | 0.74 | 1.43 | 0.23 | 0.39 | 0.81 | 1.53 | 0.24 | 0.36 | 0.61 | 0.98 |
| AR | 0.22 | 0.40 | 0.58 | **1.00** | 0.20 | 0.34 | **0.53** | 1.12 | 0.21 | 0.34 | 0.50 | **0.82** |
| Pro-Hard | 0.21 | 0.40 | 0.74 | 1.50 | 0.21 | 0.47 | 0.90 | 2.02 | 0.18 | 0.37 | 0.72 | 1.44 |
| Pro-Soft | **0.14** | **0.31** | **0.56** | 1.10 | **0.14** | **0.30** | **0.53** | **1.10** | **0.12** | **0.26** | **0.49** | 0.83 |

respectively. From such two tables, it is shown that the proposed method obtains the lowest or the second lowest MAD for all cases. The denoising ability of JGF (Liu et al. 2013) is very poor. The performances of NLMR (Park et al. 2014), MLS (Bose & Ahuja 2006) and Guided (He et al. 2010) are similar. And they are inferior to the proposed method. TGV (Ferstl et al. 2013) provides comparable results to the proposed method in 2× and 4× cases, but it is lack of robustness in 16× case. Overall, AR (Yang, Ye, Li, Hou & Wang 2014) provides comparable results to the proposed method and obtains better performances on "Reindeer" and "Laundry" datasets. To compare results visually, Fig. 4.5 illustrates the results of depth map SR with noise upsampled by the state-of-the-art methods: NLMR, Guided, MLS, TGV and the proposed method. It is shown that there is strong noise left in the results of Guided and MLS. The TGV provides cleaner depth maps, but fails to preserve tiny structures such as sticks in the cup. Overall, the proposed method can suppress noise and protect most details. However, there are some blurry artifacts near a small part of edge in the result of the proposed method

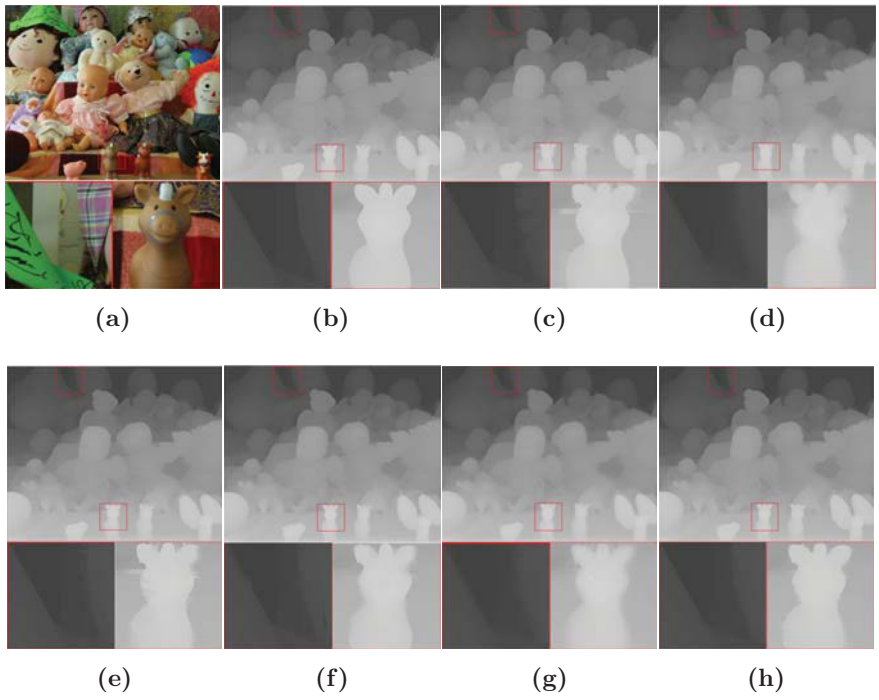


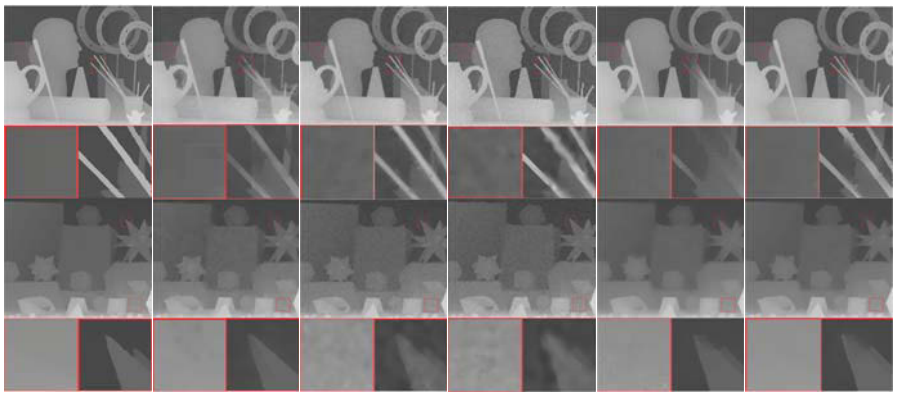

**Figure 4.4:** The visual quality comparison for depth map SR on "Dolls" dataset: (a) color image, (b) depth ground truth, depth maps are upsampled (8×) by (c) NLMR, (d) MLS, (e) JGF, (f) AR, (g) TGV, (h) the proposed method.

on "Moebius" dataset. The main reason is that this weak edge in the color image cannot be detected by Canny detector with the predefined thresholds. On the contrary, the corresponding region on Ground truth depth map has strong depth discontinuity. According to the proposed method, it is a case of definite inconsistency that color image is not adopted as the guidance for upsampling. Therefore, it may lead to blurry artifacts in the case of higher upsampling factors, e.g. 8×, 16×. However, NLMR may performs better in such region due to the higher level cues, e.g. segmentation and/or edge saliency map. But these high level cues are not stable. Actually, it is shown that its results have clear noise left and are also seriously polluted by blurring depth edges and texture-copying artifacts.

3) Degradation by structural errors and random missing

Table 4.7: Quantitative Upsampling Results (in MAD) on Noisy Middlebury Datasets "Art", "Book" and "Moebius"

| Methods \ Datasets | Art | | | | Book | | | | Moebius | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 2× | 4× | 8× | 16× | 2× | 4× | 8× | 16× | 2× | 4× | 8× | 16× |
| Bicubic | 3.52 | 3.84 | 4.47 | 5.72 | 3.30 | 3.37 | 3.51 | 3.82 | 3.28 | 3.36 | 3.50 | 3.80 |
| MLS | 1.43 | 1.95 | 3.37 | 4.67 | 0.81 | 1.39 | 2.68 | 3.21 | 0.87 | 1.40 | 2.65 | 3.16 |
| Guided | 1.49 | 1.97 | 3.00 | 4.91 | 0.80 | 1.22 | 1.95 | 3.04 | 1.18 | 1.90 | 2.77 | 3.55 |
| NLMR | 1.69 | 2.40 | 3.60 | 5.75 | 1.12 | 1.44 | 1.81 | 2.59 | 1.13 | 1.45 | 1.95 | 2.91 |
| JGF | 2.36 | 2.74 | 3.64 | 5.46 | 2.12 | 2.25 | 2.49 | 3.25 | 2.09 | 2.24 | 2.56 | 3.28 |
| TGV | 0.82 | 1.26 | 2.76 | 6.87 | 0.50 | 0.74 | 1.49 | 2.74 | 0.56 | 0.89 | 1.72 | 3.99 |
| AR | 0.76 | **1.01** | **1.70** | 3.05 | 0.47 | 0.70 | 1.15 | 1.81 | 0.46 | 0.72 | **1.15** | 1.92 |
| Pro-Soft | **0.74** | 1.02 | 1.72 | **3.01** | **0.45** | **0.66** | **1.07** | **1.80** | **0.45** | **0.68** | 1.18 | **1.85** |



**(a)** **(b)** **(c)** **(d)** **(e)** **(f)**

**Figure 4.5:** The visual quality comparison for depth map SR with noise on "Art" and "Moebius" datasets. (a) ground truth, depth maps are upsampled (8×) by (b) NLMR, (c) Guided, (d) MLS, (e) TGV, (f) the proposed method.

To quantitatively test the effectiveness of depth map completion, the proposed method uses the datasets created by AR (Yang, Ye, Li, Hou & Wang 2014) which manually adds some holes in the ground truth of Middlebury datasets. The holes consist of structural errors and random missing which are generated near depth edges and in smooth regions respectively. The experimental results are listed in Tab. 4.9 compared with 5 benchmark and

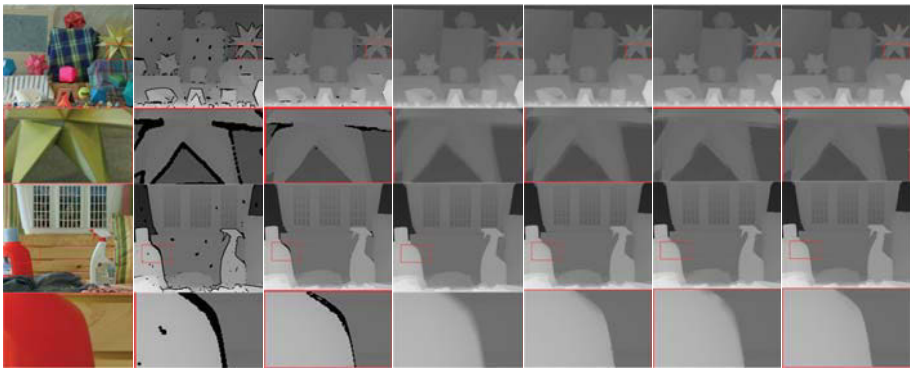
Table 4.8: Quantitative Upsampling Results (in MAD) on Noisy Middlebury Datasets "Reindeer", "Laundry" and "Dolls"

| Datasets / Methods | Reindeer | | | | Laundry | | | | Dolls | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 2× | 4× | 8× | 16× | 2× | 4× | 8× | 16× | 2× | 4× | 8× | 16× |
| Bicubic | 3.39 | 3.52 | 3.82 | 4.45 | 3.35 | 3.49 | 3.77 | 4.35 | 3.28 | 3.34 | 3.47 | 3.72 |
| MLS | 0.92 | 1.49 | 2.86 | 3.53 | 0.94 | 1.53 | 2.83 | 3.58 | 0.81 | 1.34 | 2.57 | 3.09 |
| Guided | 1.29 | 1.99 | 2.99 | 4.14 | 1.28 | 2.05 | 3.04 | 4.10 | 1.19 | 1.94 | 2.80 | 3.50 |
| NLMR | 1.20 | 1.60 | 2.40 | 3.97 | 1.28 | 1.63 | 2.20 | 3.34 | 1.14 | 1.54 | 2.07 | 3.02 |
| JGF | 2.18 | 2.40 | 2.89 | 3.94 | 2.16 | 2.37 | 2.85 | 3.90 | 2.09 | 2.22 | 2.49 | 3.25 |
| TGV | 0.59 | 0.84 | 1.75 | 4.40 | 0.61 | 1.59 | 1.89 | 4.16 | 0.66 | 1.63 | 1.75 | 3.71 |
| AR | **0.48** | **0.80** | **1.29** | **2.02** | **0.51** | **0.85** | 1.30 | **2.24** | 0.59 | 0.91 | 1.32 | 2.08 |
| Pro-Soft | 0.53 | 0.82 | 1.31 | 2.14 | 0.54 | 0.89 | **1.24** | 2.33 | **0.52** | **0.84** | **1.25** | **1.92** |

Table 4.9: Quantitative Completion Results (in MAD) on Synthetic Datasets

| Methods / Datasets | Art | Book | Moebius | Reindeer | Laundry | Dolls |
|---|---|---|---|---|---|---|
| Bicubic | 0.90 | 0.61 | 0.66 | 0.95 | 0.91 | 0.76 |
| MLS | 0.91 | 0.58 | 0.72 | **0.68** | 0.72 | 0.82 |
| JBF | 0.84 | 0.63 | 0.69 | 0.92 | 0.88 | 0.76 |
| Guided | 1.20 | 0.63 | 0.67 | 0.96 | 0.94 | 0.76 |
| AR | **0.58** | 0.53 | 0.60 | **0.68** | 0.75 | 0.69 |
| Pro-Soft | 0.60 | **0.52** | **0.56** | 0.70 | **0.71** | **0.68** |

the state-of-the-art methods. As shown in the Tab. 4.9, the proposed method obtains the lowest MAD in four datasets and the sub-optimal results in the rest two datasets, which proves the effectiveness. The visual results of the proposed method, Guided (He et al. 2010), JBF (Kopf et al. 2007) and AR (Yang, Ye, Li, Hou & Wang 2014) are shown in Fig. 4.6. Although all the methods obtain good results in depth map completion, the results of the proposed method can provide more accurate depth edges which is shown in the highlighted regions.



(a) (b) (c) (d) (e) (f) (g)

**Figure 4.6:** The visual quality comparison for depth map completion on "Moebius" and "Laundry" datasets with structural errors and random missing: (a) color images, (b) degraded depth maps, (c) ground truth, depth completed by (d) Guided, (e) JBF, (f) AR and (g) the proposed method.

### Depth Enhancement Experiments Using Real Datasets

The proposed method is also tested on ToF-Mark datasets (Ferstl et al. 2013) and NYU datasets (Silberman et al. 2012) corresponding to two types of depth sensors respectively (i.e., ToF depth sensor and Structured-light depth sensor). The experiments are to prove that the proposed method can reconstructed high quality depth maps from low quality depth maps captured by different type of sensors.

1) Experiments on ToF-Mark datasets

The proposed method is assessed on ToF-Mark datasets consisting of three RGB-D datasets, "Books, "Shark and "Devil, with ground-truth depth maps. The resolution of the original depth maps is 120×160, and the corresponding intensity images are the size of 610×810. The suggested upsampling factor is approximately 6.25× (Ferstl et al. 2013). Tab. 4.10 illustrates quantitative comparison results with optimal and suboptimal results marked in bold and underlined respectively. The upsampling errors are computed by MAD in mm unit. The proposed method obtains the lowest or second low-

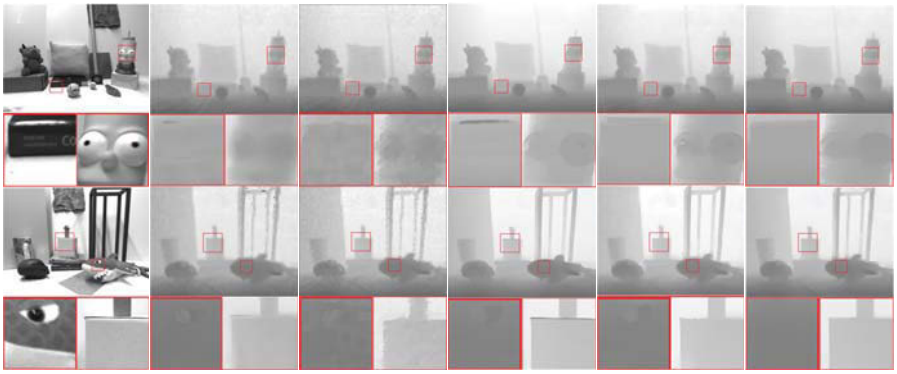
Table 4.10: Quantitative Upsampling Results (in MAD) on ToF-Mark Datasets

| Datasets | Bicubic | OMRF | Guided | MLS | JBU | JGF | NLMR | TGV | AR | Pro-Soft |
|---|---|---|---|---|---|---|---|---|---|---|
| Books | 16.23 | 13.87 | 14.51 | 14.50 | 14.78 | 17.39 | 14.31 | **11.90** | 12.45 | 12.23 |
| Shark | 17.78 | 16.07 | 16.62 | 16.26 | 17.15 | 18.17 | 15.88 | 14.47 | 14.71 | **14.14** |
| Devil | 16.66 | 15.36 | 24.97 | 14.97 | 25.46 | 19.02 | 15.36 | 13.90 | 13.83 | **13.71** |

est MAD error for all the three datasets compared with other 9 benchmark and the state-of-the-art methods. Fig. 4.7 shows the visual depth enhancement results of the proposed method against the 4 state-of-the-art methods (MLS (Bose & Ahuja 2006), JGF (Liu et al. 2013), TGV (Ferstl et al. 2013) and AR (Yang, Ye, Li, Hou & Wang 2014)). It is observed that the results of MLS and JGF still contain considerable amount of noise due to the limited denoising ability, while the depth enhanced by TGV, AR and the proposed method are much better. However, the results of TGV and AR introduce texture-copying artifacts in some smooth regions, e.g., the eye of the fish in "Shark" dataset highlighted by red square. Results of the proposed method do not have such texture-copying artifacts. In addition, the edge of the rectangular box in "Shark" dataset highlighted by red square is more accurate in the result of the proposed method than that of others, which proves that the proposed method can efficiently preserve depth edges.

2) Experiments on NYU datasets

By using NYU datasets (Silberman et al. 2012) in which the depth maps are captured by structured-light depth sensors, the proposed method is evaluated for depth map completion, compared with 4 state-of-the-art methods: AR (Yang, Ye, Li, Hou & Wang 2014), MLS (Bose & Ahuja 2006), JBU (Kopf et al. 2007) and Colorization (Levin, Lischinski & Weiss 2004). Fig. 4.8 shows the comparison of depth map completion results. From these highlighted regions, it is shown that the existing methods (e.g., AR and MLS) suffer from texture-copying artifacts (e.g. highlighted in the second row). By contrast, there are no such artifacts in results of the propose method. In term of pre-



(a) (b) (c) (d) (e) (f)

**Figure 4.7:** The visual quality comparison for depth map SR on "Devil" and "Shark" datasets: (a) color images, depth maps are upsampled by (b) MLS, (C) JGF, (d) TGV, (e) AR and (f) the proposed method.

serving depth edges, AR performs best in these existing methods, but it is inferior to the proposed method (e.g. highlighted in the fourth row). Therefore, the proposed method can provide more robust results of depth map completion than the state-of-the-art methods in real scenes.

### Experiments on Tackling both Depth Map SR and Depth Map Completion

In the previous experiments, the performances of the proposed method on depth map SR and depth map completion are shown independently. In order to further verify the robustness of the proposed method, the experiments in this part are to tackle an extremely difficult case in which the proposed method is going to upsample the LR depth map (i.e., 4× upsampling factor) and complete holes simultaneously. NYU datasets (Silberman et al. 2012) are adopted in the experiments. Fig. 4.9 shows the results of the proposed method, compared with Colorization (Levin et al. 2004), JBU (Kopf et al. 2007) and TGV (Ferstl et al. 2013). From the highlighted regions, it is shown that the proposed method provides the best performances in holes

(a) (b) (c) (d) (e) (f) (g)

**Figure 4.8:** The visual quality comparison for depth map completion on NYU datasets: (a) color images, (b) Registered raw depth maps from Kinect v1, completed by (c) AR, (d) MLS, (e) JBU, (f) Colorization and (g) the proposed method.

filling, mitigating texture-copying artifacts and preserving depth edges. By contrast, Colorization shows texture-copying artifacts (e.g. highlighted in the second row) and blurring depth edges (e.g. highlighted in the fourth row) in such results. JBU and TGV cannot give satisfying results with holes left uncompleted.

### 4.4.6 Conclusion

This section proposes a novel color-guided depth map enhancement method via MRF optimization. The key contribution is to explicitly measure the inconsistency between the color edge map and the corresponding depth edge map in numeral way. In the following step, such quantitative measurement is embedded into MRF-based model. It adaptively controls the efforts of the guidance from the color image. The proposed fine-grained method can better mitigate texture-copying artifacts and preserve depth edges than the counterpart based on hard-decision edge inconsistency measurement. To verify the proposed method, enough experiments on Middlebury datasets, ToF-Mark datasets and NYU datasets for depth map SR and depth map completion tasks are conducted. Furthermore, the proposed method is able

(a) (b) (c) (d) (e)

**Figure 4.9:** The visual quality comparison for depth map enhancement with the complex degradation (downsampling and depth values missing) on NYU datasets: (a) color images with the corresponding LR depth maps shown on the upper left corner, depth maps are enhanced by (b) Colorization, (c) JBU, (d) TGV and (e) the proposed method.

to handle both depth map SR and depth map completion simultaneously. All the experimental results prove the improved performances of the proposed method when compared with the state-of-the-art methods.

Although the promising depth enhancement results can be obtained by using the proposed method, there are some failure cases when the quality of original depth map is too low (e.g., 8×, 16× upsampling factor). A corresponding failure case is shown in the result on "Moebius" dataset in Fig. 4.5. Next section will propose a new method to improve robustness when the quality of input depth map is very low.

## 4.5 Minimum Spanning Forest Embedded with Soft Edge Inconsistency Measurement for Guided Depth Map Enhancement

To better preserve the structure on depth map, a structural scheme for affinity computing in the regularization term is proposed in this section. Affinities are computed more precisely in a space which consists of multiple Minimum Spanning Trees (Forest) than the existing methods. It is calculated based on paths which travel inside a Minimum Spanning Tree (MST) or across adjacent MSTs. The local depth structure can be represented by such paths. In addition, soft edge inconsistency measurement which is proposed in previous section is embedded into the edge weights inside each MST to significantly mitigate texture-copying artifacts. For brief, soft edge inconsistency measurement is the same as inconsistency model in this section.

### 4.5.1 Modification on MRF Energy Function

The data term is defined based on M-estimator Huber (Huber 1973) which is a trade-off between L1-norm and L2-norm:

$$E_{data}\left(\mathbf{D}\right) = \left\|\mathbf{D_{sub}} - \mathbf{O}\right\|_{\mathbf{Huber}} \tag{4.14}$$

$$\left\|\mathbf{x}\right\|_{\mathbf{Huber}} = \begin{cases} \sum\limits_{i} \frac{\mathbf{x}_i^2}{2\epsilon} & |\mathbf{x}_i| \leqslant \epsilon, \\ \sum\limits_{i} \epsilon\left(|\mathbf{x}_i| - \frac{\epsilon}{2}\right) & |\mathbf{x}_i| > \epsilon \end{cases} \tag{4.15}$$

where $\mathbf{O}$ is a set of observed depth values which are captured by sensors directly. $\mathbf{D_{sub}}$ is the subset of $\mathbf{D}$. It consists of depth of pixels which have observe values.

Furthermore, compared with Eq. (4.3) that the affinity $\lambda_s^{pq}$ is computed by the differences of depth and color directly, this section redefines $\lambda_s^{pq}$ based on

the distance in minimum spanning forest (MSF) which is shown as Eq. (4.16).

$$\lambda_s^{pq} = e^{\frac{-dist(p,q)}{\delta}} \tag{4.16}$$

where $\delta$ controls bandwidth of the exponential function, $dist\,(p,q)$ stands for the distance between pixels $p, q$ in the MSF. In the following parts, the details of the proposed method will be explained.

### 4.5.2 The Proposed Method

The existing methods (Diebel & Thrun 2005, Lu et al. 2011, Park et al. 2014, Yang, Ye, Li, Hou & Wang 2014, Liu et al. 2017) including the proposed methods in the previous subsections calculate guidance affinities of regularization term via non-structural scheme which is only based on color and depth differences between the pixel and its neighbor pixels. Such scheme ignores the local structure on the depth map. Therefore, it may over-smooth depth edges on the enhanced depth map (e.g., 16× for depth SR). By contrast, it is observed that tree filter (Zhang, Dai, Xiang & Zhang 2015, Bao et al. 2014) can provide more robust performance in preserving edges. It has been used in many computer vision tasks, e.g., structure-preserving smoothing (Zhang et al. 2015, Bao et al. 2014), stereo matching (Yang 2015). In these methods, MST is used to automatically drag away two dissimilar pixels that are close to each other in the spatial domain, which makes the tree distance be an edge-aware metric. However, global MST faces the problem of leaking edges (Zhang et al. 2015, Bao et al. 2014). That is, one MST connects all pixels in an image together and aggregates the smoothing effects globally along the MST. In such construction, connections will inevitably go cross some strong edges, and these edges will be corrupted. The proposed method computes the anisotropic guidance affinity $\lambda_s^{pq}$ in the regularization term based on tree distance between $p$ and $q$. And in order to avoid constructing a MST across depth edges, which leads to leaking-edge artifacts, motivated by (Zhang et al. 2015), the proposed method constructs a MST for each super-pixel on the color image generated by over-segmentation. Be-

cause the color image in our case is high-quality and the pixels inside the same super-pixel have similar characteristics, the tree distance calculation is more reliable in the region of super-pixel. Furthermore, due to over-segmentation, the pixel and its neighboring pixels which locate on the adjacent MSTs may have similar depth values. So the edge links between adjacent MSTs should also be calculated in order to consider these situations. The adjacent MSTs join together and become so called Minimum Spanning Forest (MSF).

If guidance affinities are merely computed based on registered color image, it leads to texture-copying artifacts and blurring depth edges due to the edge inconsistency between depth map and the registered color image. Motivated by the proposed inconsistency model (Zuo, Wu, Zhang & An 2018) which shows strong ability in mitigating texture-copying artifacts, edge weights inside each MST in the proposed MSF are computed based on such model. These two parts of contribution complement each other to obtain robust depth enhancement performance.

### The Construction of MSF

SLIC super-pixel segmentation algorithm (Achanta, Shaji, Smith, Lucchi, Fua & Süsstrunk 2012) is adopted to over-segment the color image which avoids constructing a minimum spanning tree across depth edges to mitigate leaking-edge artifacts (Zhang et al. 2015, Bao et al. 2014). This state-of-the-art segmentation method yields adherence to the major image edges and the complexity is linear. In each segment, an 8-connected weighted subgraph $\mathbf{G_{sub}}(\mathbf{V}, \mathbf{E}, \mathbf{W})$ is generated. All the pixels in this segment are nodes $\mathbf{V}$. $\mathbf{E}$ consists of all the edges between nodes. $\mathbf{W}$ is the weight set of edges which is defined as Eq. (4.17) in (Zhang et al. 2015).

$$\mathbf{W}(p, q) = |\nabla_{color}^{pq}| \tag{4.17}$$

where $\nabla_{color}^{pq}$ is the color difference between $p, q$. Since the color image and the depth map have different texture patterns, such configuration leads to texture-copying artifacts and blurring depth edges. The proposed method

explicitly embeds edge inconsistency measurement into MST construction to address these problems which will be explained in the next subsection.

So inside each MST, currently, each edge weight between two adjacent nodes is the color similarity of them as Eq. (4.17). Based on such subgraphs, a MST can be computed on each super-pixel by removing redundant edges. Due to over-segmentation, the pixel and its neighboring pixels which locate on the adjacent MSTs may still have similar depth values. So the edges linking adjacent MSTs should also be calculated in order to consider these situations. To be consistent with the weight of the edges inside each MST, the weight of the edge linking adjacent MSTs is computed by the color and the depth differences to mitigate texture-copying artifacts. Specially, such edge defined by the pixels with similar color and depth which locate on the adjacent MST can extend MST on super-pixel to MSF on the whole image without across edges. It can be considered as the supervision to overcome leaking-edge problem. The similar setting is also used in (Zhang et al. 2015) which only considers the color similarity to perform edge-aware smoothing. In fact, the proposed MSF can be treated as a distance space based on the image content. The spatial factor is included in the path between pixels in the MSF. Compared with non-structural Euclidean distance space which is used in two separate kernels of bilateral filter (Tomasi & Manduchi 1998), the distance computed in the proposed MSF presents the structure of the image. The proposed method models each pixel $s$ as a two-dimension point $\mathbf{Pt_s}\left(c_s, d_s\right)$ which includes its color value $c_s$ and the depth value $d_s$. And the distance metric between two pixels is defined as L1-norm. The optimal node pair $(p^\star, q^\star)$ between the adjacent super-pixels $Sup_\beta$ and $Sup_\gamma$ is defined as Eq. (4.18).

$$(p^\star, q^\star) = \min_{\substack{p \in Sup_\beta \\ q \in Sup_\gamma}} \left\|\mathbf{Pt_p} - \mathbf{Pt_q}\right\|_1 \tag{4.18}$$

If exhaustive search is applied, the complexity is $O\left(m_1 m_2\right)$ where $m_1, m_2$ are the numbers of nodes in their MSTs respectively. It is time-consuming when the size of super-pixel is large. The proposed method adopts the more

efficient divide-and-conquer algorithm (Cormen, Leiserson, Rivest & Stein 2001) to address this problem which reduces the computation complexity to $O\left((m_1 + m_2)\, log\,(m_1 + m_2)\right)$.

Consequently, the distance between two neighboring nodes in the regularization term is computed as follows; 1) if $p, q$ are located in the same super-pixel, the distance between them can be computed through standard tree distance along the path on the MST.

$$dist\,(p, q) = \sum_{i=0}^{n} \mathbf{W}\,(p_i, p_{i+1}) \tag{4.19}$$

where $(p_i, p_{i+1})$ are adjacent nodes along the path. Such tree distance is efficiently computed based on Lowest Common Ancestor method (LCA) (Bender & Farach-Colton 2000). 2) Otherwise, the distance is computed in three parts;

$$\begin{aligned} dist\,(p, q) =& dist\,(p, p^\star) + dist\,(q, q^\star) \\ &+ 0.5 \times \left( \left| \nabla_{color}^{p^\star q^\star} \right| + \left| \nabla_{depth}^{p^\star q^\star} \right| \right) \end{aligned} \tag{4.20}$$

where $p^\star, q^\star$ are two closest node pair inside adjacent MSTs respectively. They are closest considering both color similarity and depth similarity as the last term on the right side of Eq. (4.20). $\nabla_{color}^{p^\star q^\star}$ and $\nabla_{depth}^{p^\star q^\star}$ represent color and depth differences between the pixels $p^\star$ and $q^\star$ on the guided color image and the coarsely interpolated depth map respectively. $(p, p^\star)$ and $(q, q^\star)$ are pairs of pixels located in the same MST.

**Embedding Edge Inconsistency Measurement into MSF**

The edge inconsistency measurement model has strong ability in mitigating texture-copying artifacts. For content integrity, this part summaries it as follows; it is a bi-direction evaluation by swapping the roles of color edge map and depth edge map. In each direction, the best matching pairs of edge pixels are determined through a MRF optimization. The data term implies local structure information which is based on Minimum Weighted Bipartite Matching (Kuhn 1955) and regularization term implies global structure

information. The cost of the optimal matching edge pixels is the edge inconsistency measurement which is in the range of [0, 1]. For more detail, please refer to Section 4.4.2. Motivated by this model, the proposed method embeds it into the MST construction to mitigate texture-copying artifacts.

In the proposed method, $\boldsymbol{\alpha}$ is defined as the set of confidence values for pixels which is computed according to Eq. (4.10) in Section 4.4.2. The smaller value $\boldsymbol{\alpha}$ is, the more edge consistency is represented. More specifically, the proposed method modifies the definition of $\mathbf{W}$ in Eq. (4.17) for the edge $(p, q)$ as;

$$\begin{aligned}\mathbf{W}'(p,q) = &|\nabla^{pq}_{color}| \times (1-\alpha_{pq}) \\ &+ \left|\nabla^{pq}_{depth}\right| \times \alpha_{pq}\end{aligned} \tag{4.21}$$

where $\alpha_{pq}$ is the confidence value for pixel pair $p, q$ which is defined as $\alpha_{pq} = \max(\boldsymbol{\alpha}(p), \boldsymbol{\alpha}(q))$ for better preserving depth edges. $\boldsymbol{\alpha}(p), \boldsymbol{\alpha}(q)$ are the confidence values for $p$ and $q$ respectively. When the color edge map is more consistent with the depth edge map ($\alpha_{pq}$ is more close to zero), $\nabla^{pq}_{color}$ is able to play a more important role in computing edge weights inside the MST, and vice versa.

In this section, the newly proposed MST edge weights $\mathbf{W}'$ defined in Eq. (4.21) will replace the original $\mathbf{W}$ in Eq. (4.17), Eq. (4.19) and Eq. (4.20). Compared with the previous MRF-based depth enhancement methods which do not use structure information in regularization term calculation, the proposed method in this section can better preserve depth edges by using the distance on MSF defined in Eq. (4.19) and Eq. (4.20). In addition, texture-copying artifacts are significantly mitigated through proposed improved MST construction with explicitly embedding the edge inconsistency measurement model.

### Bandwidth Adaptation for Affinities Computing in MRF

In the work, it is observed that bandwidth $\delta$ in Eq. (4.16) can affect the performance of model. For instance, when the two pixels applied into regularization term of MRF have clearly different depth values, small bandwidth

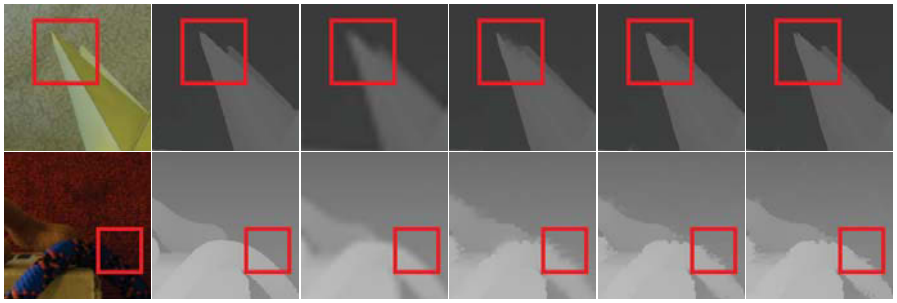
$\delta$ can provide better performance. Therefore, if the depth difference between each neighbor pixel pair in the regularization term is known, bandwidth $\delta$ can be adaptively chosen as a prior when reconstructing the high-quality depth map. This prior can preserve depth edges and mitigate the artifacts caused by noise and texture copying. In this section, the coarsely interpolated depth map $\hat{\mathbf{D}}$ is used to estimate such prior which adaptively controls $\delta$.

Because of the low quality of $\hat{\mathbf{D}}$, the edge locations shift from the real ones. Therefore, it is inaccurate to estimate the depth difference between the pixel pair $p, q$ by directly computing on themselves. The proposed method defines two sub-regions ($11 \times 11$ in this section) which are centered by $p, q$ respectively to form pixel sets $\mathbf{S_p}, \mathbf{S_q}$. The maximum absolute difference between these sets $AD_{max}$ can be used to analyse the potential prior for $p, q$. If $AD_{max}$ is large, it indicates that pixel pair $p, q$ is close to depth edges nearby and the corresponding $\delta$ should be assigned a small value to better preserve depth edges. By contrast, if $AD_{max}$ is small, it means this pixel pair $p, q$ is located at a smooth region. In this case, a large value should be chosen for $\delta$ to smooth noise and further mitigate texture-copying artifacts.

The simple implementation of computing $AD_{max}$ is to compute the differences for all the elements in two sets $\mathbf{S_p}, \mathbf{S_q}$ with the complexity of $O(t^2)$ where $t$ is the cardinal number for each set. Such implementation is time-consuming. A more efficient method is proposed, which has the complexity of $O(t)$. Firstly, the minimum and the maximum depth values (i.e., $max_p, min_p, max_q, min_q$) are computed for sets $\mathbf{S_p}, \mathbf{S_q}$ respectively. Then $AD_{max}$ can be calculated as below.

$$AD_{max} = \max\left(\left|max_p - min_q\right|, \left|max_q - min_p\right|\right) \tag{4.22}$$

**Proof 4.1** *$0 \leqslant min_p \leqslant e_p \leqslant max_p$, $0 \leqslant min_q \leqslant e_q \leqslant max_q$, where $e_p, e_q$ are arbitrary element in sets $\mathbf{S_p}, \mathbf{S_q}$ respectively. Then, $min_p - max_q \leqslant e_p - e_q \leqslant max_p - min_q$. Therefore, $|e_p - e_q| \leqslant \max\left(\left|max_p - min_q\right|, \left|max_q - min_p\right|\right)$*



**(a)** **(b)** **(c)** **(d)** **(e)** **(f)**

**Figure 4.10:** Visual comparison of upsampled (8×) depth maps, (a) color patches, (b) ground truth depth patches, LR depth maps are upsampled by: (c) bicubic, (d) inconsistency model only, (e) MSF+inconsistency model, (f) MSF+inconsistency model+parameter adaption.

Since the situations of noise are diverse, it is difficult to model the mapping function between $\delta$ and $AD_{max}$ as a fixed model (e.g., linear model). In the proposed method, $AD_{max}$ is computed for each pixel pair in the regularization term and classify all the pixel pairs into three categories using double thresholds $\theta_{low}, \theta_{high}$. Such categories represent near strong edge regions, near weak edge regions and smooth regions. For each type of regions, different $\delta$ values are chosen which is defined as Eq. (4.23). The optimal values of $\delta$ is determined on "Laundry" dataset and applied to the rest experiments.

$$\delta = \begin{cases} 2, & AD_{max} > \theta_{high} \ (strong\ edge) \\ 10, & AD_{max} < \theta_{low} \ (smooth\ region) \\ 4, & Others \ (weak\ edge) \end{cases} \tag{4.23}$$

Fig. 4.10 and Fig. 4.11 show subjective and objective comparison of the enhanced depth maps between non-structural scheme embedded with inconsistency model (Zuo et al. 2018), MSF-based scheme embedded with inconsistency model and the proposed method on the noise-free Middlebury datasets. Progressive improvement can be observed when the proposed MSF and parameter adaption are used respectively. More improved results are shown in the experimental subsection.

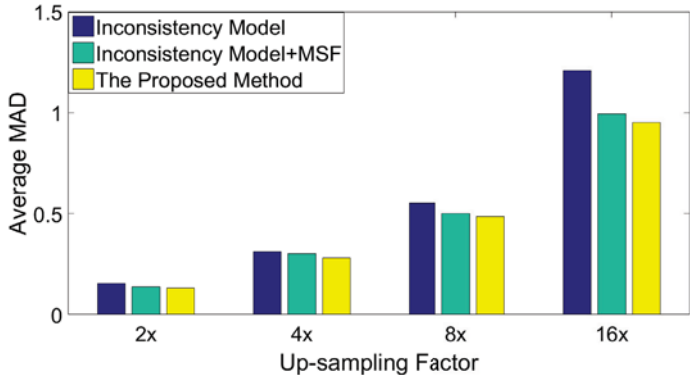



**Figure 4.11:** Quantitative comparison between the results enhanced by Inconsistency model only, Inconsistency model+MSF and The proposed method in terms of average MAD on Middleburry datasets.

### 4.5.3 Experimental Results

The performance of the proposed method is evaluated for depth map SR and depth map completion on various datasets. After the investigation of influence of the parameters, the proposed model is firstly tested for depth map SR on the synthetic datasets (*Middlebury Datasets [Online]* n.d., Yang, Ye, Li, Hou & Wang 2014) and the real ToF-Mark datasets (Ferstl et al. 2013) which captured by the ToF sensor. Then depth map completion experiments are conducted on the synthetic datasets (Yang, Ye, Li, Hou & Wang 2014) and the real datasets (Silberman et al. 2012) obtained by the Kinect sensor. Finally, the average running time comparison is shown.

#### Influence of Parameters

Firstly, the influence of some key parameters, including the amount of super-pixel and the balance Factor $\lambda$ in Eq. (4.3) are investigated.

1) The amount of super-pixels

The amount of super-pixels is modified and remain other parameters unchanged ($\epsilon = 10$ in Eq. (4.15), $\theta_{low} = 5, \theta_{high} = 30$ and $\delta = [2, 10, 4]$ for corresponding cases in Eq. (4.23)). And we adopt LBFGS to remember 10 of the previous optimization steps in order to construct an approximate Hes-

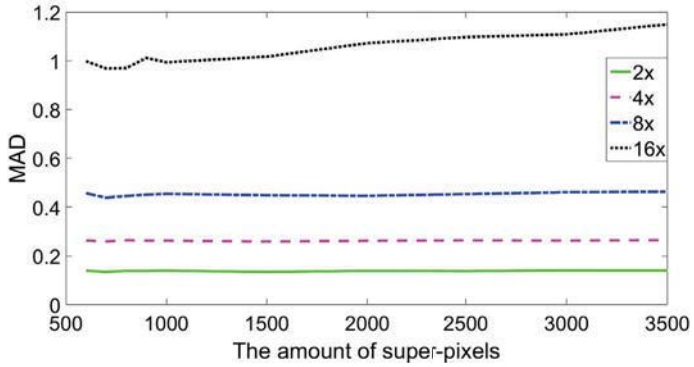



**Figure 4.12:** The influence of the amount of the super-pixels

sian matrix. Fig. 4.12 provides an illustration of the depth map SR result of "Laundry" dataset in the case of 2×, 4×, 8× and 16×. It can be noticed that the effects caused by different amount of super-pixels are not significant, especially for small upsampling factors (i.e., 2×, 4×). The optimized value is about 700 which provides robust results for all cases in this experiment. It is fixed to 700 in the following experiments.

2) The Balance Factor $\lambda$ in Eq. (4.3)

The similar analysis as that in Section 4.4.5 can be shown. In this work, it is fixed to 0.01 for all upsampling factors when upsampling depth maps without adding noise. For depth map SR with adding noise, $\lambda$ is set to 5, 3.3, 2.5 and 1.43 for 2×, 4×, 8× and 16× respectively. For depth map completion, it is set $\lambda$ to 0.1 and 10 for all the depth map completion experiments on synthetic datasets (Yang, Ye, Li, Hou & Wang 2014) and real datasets (Silberman et al. 2012) respectively. It should be noted that the proposed method applies the value of $\lambda$ adjusted based on the "Laundry" dataset for depth map SR and noise-free completion. For noisy depth map completion on the NYU datasets (Silberman et al. 2012), only one image pair is used to tune $\lambda$.

**Experiments on Depth SR**

In this part, the experiments for depth map SR are evaluated on synthetic datasets (*Middlebury Datasets [Online]* n.d., Yang, Ye, Li, Hou & Wang 2014) and real datasets (Ferstl et al. 2013) as well.

**Depth SR for synthetic datasets** The experiments on synthetic datasets which includes two kinds of degradation: downsampling and downsampling with adding noise are shown in this part.

1) Degradation by downsampling

Firstly, the experiments are run on noise-free datasets which are filled ground truth data (*Middlebury Datasets [Online]* n.d.) downscaled by nearest neighbor interpolation. The proposed method (Pro-MSF) is tested for multiple upsampling factors (i.e., 2×, 4×, 8×, 16×) and compared with 12 benchmark and the state-of-the-art methods: Bicubic interpolation (bicubic), Original MRF-based method (OMRF) (Diebel & Thrun 2005), Joint bilateral upsampling (JBU) (Kopf et al. 2007), Improved JBU embedded edge inconsistency measurement model (IMJBU), Spatial-depth super resolution for range images (JBUV) (Yang et al. 2007), Guided image filtering (Guided) (He et al. 2010), Edge-weighted NLM-regularization (NLMR) (Park et al. 2014), Joint geodesic filtering (JGF) (Liu et al. 2013), Moving least squares filter (MLS) (Bose & Ahuja 2006), Total generalized variation (TGV) (Ferstl et al. 2013), Auto-regression model (AR) (Yang, Ye, Li, Hou & Wang 2014) and Pro-Soft proposed in Section 4.4. Moreover, it is realized that OMRF and JBUV did not report the experimental results on the datasets of "Reindeer", "Laundry" and "Dolls".

Tab. 4.11 and Tab. 4.12 show the upsampling results under four different upsampling factors with optimal and suboptimal results marked in bold and underlined respectively. Overall, the proposed method obtains the lowest MAD for most cases. Thanks to the MSF-based guidance affinities computing scheme, the proposed method reaches the lowest MAD on all datasets for 16×, 5 datasets for 8×, 5 datasets for 4× and 4 datasets for 2×. For the

Table 4.11: Quantitative Upsampling Results (in MAD) on Noise-free Middlebury Datasets "Art", "Book" and "Moebius"

| Datasets / Methods | Art | | | | Book | | | | Moebius | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 2× | 4× | 8× | 16× | 2× | 4× | 8× | 16× | 2× | 4× | 8× | 16× |
| Bicubic | 0.48 | 0.97 | 1.85 | 3.59 | 0.13 | 0.29 | 0.59 | 1.15 | 0.13 | 0.30 | 0.59 | 1.13 |
| OMRF | 0.59 | 0.96 | 1.89 | 3.78 | 0.21 | 0.33 | 0.61 | 1.20 | 0.24 | 0.36 | 0.65 | 1.25 |
| JBUV | 0.55 | 0.68 | 1.44 | 3.52 | 0.29 | 0.44 | 0.62 | 1.45 | 0.38 | 0.46 | 0.67 | 1.10 |
| JBU | 0.45 | 0.85 | 1.68 | 3.35 | 0.17 | 0.36 | 0.74 | 1.56 | 0.18 | 0.37 | 0.76 | 1.46 |
| IMJBU | 0.43 | 0.83 | 1.62 | 3.26 | 0.16 | 0.34 | 0.72 | 1.47 | 0.17 | 0.36 | 0.74 | 1.39 |
| Guided | 0.63 | 1.01 | 1.70 | 3.46 | 0.22 | 0.35 | 0.58 | 1.14 | 0.23 | 0.37 | 0.59 | 1.16 |
| NLMR | 0.41 | 0.65 | 1.03 | 2.11 | 0.17 | 0.30 | 0.56 | 1.03 | 0.18 | 0.29 | 0.51 | 1.10 |
| JGF | 0.29 | 0.47 | 0.78 | 1.54 | 0.15 | 0.24 | 0.43 | 0.81 | 0.15 | 0.25 | 0.46 | 0.80 |
| TGV | 0.45 | 0.65 | 1.17 | 2.30 | 0.18 | 0.27 | 0.42 | 0.82 | 0.18 | 0.29 | 0.49 | 0.90 |
| MLS | 0.27 | 0.68 | 1.04 | 2.20 | 0.16 | 0.26 | 0.48 | 1.16 | 0.15 | 0.25 | 0.49 | 0.93 |
| AR | **0.18** | 0.49 | **0.64** | 2.01 | 0.12 | 0.22 | 0.37 | 0.77 | **0.10** | **0.20** | 0.40 | 0.79 |
| Pro-Hard | 0.40 | 0.56 | 1.03 | 2.38 | 0.14 | 0.27 | 0.48 | 0.92 | 0.15 | 0.30 | 0.62 | 1.20 |
| Pro-Soft | **0.18** | **0.45** | 0.71 | 1.97 | 0.10 | 0.20 | 0.37 | 0.74 | **0.10** | **0.20** | **0.39** | 0.80 |
| Pro-MSF | 0.19 | 0.46 | 0.69 | **1.43** | **0.09** | **0.19** | **0.36** | **0.69** | 0.11 | 0.21 | **0.39** | **0.78** |

rest cases, the proposed method achieves the performance on top rank 3. Fig. 4.13 shows the experimental results of 8× upsampled depth maps for "Laundry" and "Dolls" datasets compared with 4 state-of-the-art methods: JGF (Liu et al. 2013), TGV (Ferstl et al. 2013), NLMR (Park et al. 2014) and AR (Yang, Ye, Li, Hou & Wang 2014). It can be observed that TGV (Ferstl et al. 2013) severely suffered from texture-copying artifacts. In addition, from the highlighted regions, some artifacts near depth edges appear on the results of JGF, NLMR, TGV and AR. The results of the proposed method are closest to the ground truth depth maps without texture-copying artifacts or blurring depth edges.

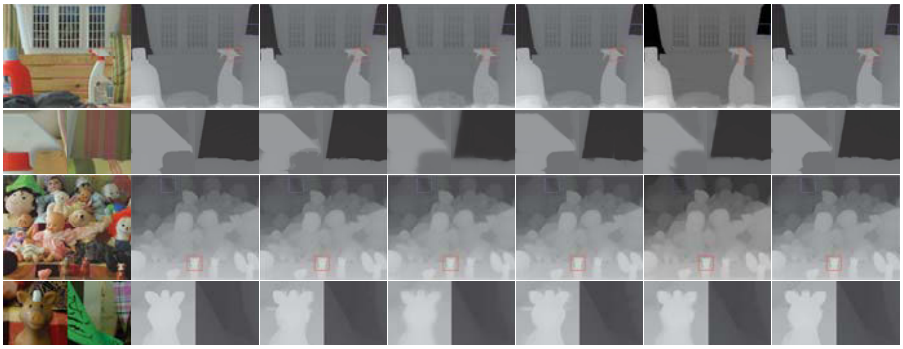
2) Degradation by downsampling with adding noise

In real situation, depth maps captured by sensors are accompanied by unavoidable noise. To simulate such cases, more experiments are run on the noisy datasets provided by AR (Yang, Ye, Li, Hou & Wang 2014). Tab. 4.13

Table 4.12: Quantitative Upsampling Results (in MAD) on Noise-free Middlebury Datasets "Reindeer", "Laundry" and "Dolls"

| Datasets / Methods | Reindeer | | | | Laundry | | | | Dolls | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 2× | 4× | 8× | 16× | 2× | 4× | 8× | 16× | 2× | 4× | 8× | 16× |
| Bicubic | 0.30 | 0.55 | 0.99 | 1.88 | 0.28 | 0.54 | 1.04 | 1.95 | 0.20 | 0.36 | 0.66 | 1.18 |
| JBU | 0.27 | 0.50 | 1.00 | 1.89 | 0.26 | 0.49 | 0.94 | 1.95 | 0.20 | 0.38 | 0.74 | 1.46 |
| IMJBU | 0.27 | 0.49 | 0.98 | 1.87 | 0.25 | 0.48 | 0.92 | 1.94 | 0.20 | 0.37 | 0.73 | 1.44 |
| Guided | 0.42 | 0.53 | 0.88 | 1.80 | 0.38 | 0.52 | 0.95 | 1.90 | 0.28 | 0.35 | 0.56 | 1.13 |
| NLMR | 0.20 | 0.37 | 0.63 | 1.28 | 0.17 | 0.32 | 0.54 | 1.14 | 0.16 | 0.31 | 0.56 | 1.05 |
| JGF | 0.23 | 0.38 | 0.64 | 1.09 | 0.21 | 0.36 | 0.64 | 1.20 | 0.19 | 0.33 | 0.59 | 1.06 |
| TGV | 0.32 | 0.49 | 1.03 | 3.05 | 0.31 | 0.55 | 1.22 | 3.37 | 0.21 | 0.33 | 0.70 | 2.20 |
| MLS | 0.32 | 0.64 | 0.74 | 1.43 | 0.23 | 0.39 | 0.81 | 1.53 | 0.24 | 0.36 | 0.61 | 0.98 |
| AR | 0.22 | 0.40 | 0.58 | 1.00 | 0.20 | 0.34 | 0.53 | 1.12 | 0.21 | 0.34 | 0.50 | 0.82 |
| Pro-Hard | 0.21 | 0.40 | 0.74 | 1.50 | 0.21 | 0.47 | 0.90 | 2.02 | 0.18 | 0.37 | 0.72 | 1.44 |
| Pro-Soft | **0.14** | 0.31 | 0.56 | 1.10 | 0.14 | 0.30 | 0.53 | 1.10 | **0.12** | 0.26 | 0.49 | 0.83 |
| Pro-MSF | **0.14** | **0.30** | **0.52** | **0.98** | **0.13** | **0.26** | **0.44** | **0.97** | **0.12** | **0.25** | **0.45** | **0.79** |

and Tab. 4.14 show the depth enhancement results of the proposed method (Pro-MSF) as well as 7 benchmark and the state-of-the-art methods with optimal and suboptimal results marked in bold and underlined respectively. From such two tables, it is shown that the proposed method obtains the lowest or the second lowest MAD for all cases. The denoising ability of JGF (Liu et al. 2013) is very poor. The performances of NLMR (Park et al. 2014), MLS (Bose & Ahuja 2006) and Guided (He et al. 2010) are similar. TGV (Ferstl et al. 2013) can obtain better results than methods above when upsampling factor is low (i.e., 2×, 4×), but it lacks robustness in the cases of large upsampling factors (i.e., 8×, 16×). Overall, AR (Yang, Ye, Li, Hou & Wang 2014) provides a little worse results compared with the proposed method. However, it obtains better performances on "Reindeer" dataset than ours. Fig. 4.14 illustrates the results of depth map SR on noisy datasets "Book" and "Moebius", the LR depth maps are upsampled by the state-of-the-art methods: Guided, JGF, TGV, NLMR and the proposed method. It



(a) (b) (c) (d) (e) (f) (g)

**Figure 4.13:** Visual comparison of upsampled (8×) depth maps on noise-free Middlebury datasets "Laundry" and "Dolls", (a) HR color images, (b) ground truth depth maps, LR depth maps upsampled by: (c) JGF, (d) TGV, (e) NLMR, (f) AR and (g) ours.

is shown that there is strong noise left on the results of Guided and JGF. Although TGV and NLMR can provide cleaner results, they severely suffer from texture-copying artifacts. Compared with these methods, the noise is significantly suppressed in the results of the proposed method which have more accurate edges without texture-copying artifacts as well.

**Depth Map SR on Real Datasets** The proposed method is tested on ToF-Mark datasets (Ferstl et al. 2013) for depth map SR to prove the robustness on real datasets captured by ToF sensors. The resolution of the LR depth maps is 120×160, and the registered intensity images are the size of 610×810. The suggested upsampling factor is approximately 6.25× (Ferstl et al. 2013).

Tab. 4.15 illustrates quantitative comparison results with optimal and suboptimal results marked in bold and underlined respectively. The upsampling errors are computed by MAD in mm unit. The results of the proposed method show the lowest MAD error for all the three datasets compared with other 11 benchmark and the state-of-the-art methods. Fig. 4.15 shows the

Table 4.13: Quantitative Upsampling Results (in MAD) on Noisy Middlebury Datasets "Art", "Book" and "Moebius"

| Datasets / Methods | Art | | | | Book | | | | Moebius | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 2× | 4× | 8× | 16× | 2× | 4× | 8× | 16× | 2× | 4× | 8× | 16× |
| Bicubic | 3.52 | 3.84 | 4.47 | 5.72 | 3.30 | 3.37 | 3.51 | 3.82 | 3.28 | 3.36 | 3.50 | 3.80 |
| MLS | 1.43 | 1.95 | 3.37 | 4.67 | 0.81 | 1.39 | 2.68 | 3.21 | 0.87 | 1.40 | 2.65 | 3.16 |
| Guided | 1.49 | 1.97 | 3.00 | 4.91 | 0.80 | 1.22 | 1.95 | 3.04 | 1.18 | 1.90 | 2.77 | 3.55 |
| NLMR | 1.69 | 2.40 | 3.60 | 5.75 | 1.12 | 1.44 | 1.81 | 2.59 | 1.13 | 1.45 | 1.95 | 2.91 |
| JGF | 2.36 | 2.74 | 3.64 | 5.46 | 2.12 | 2.25 | 2.49 | 3.25 | 2.09 | 2.24 | 2.56 | 3.28 |
| TGV | 0.82 | 1.26 | 2.76 | 6.87 | 0.50 | 0.74 | 1.49 | 2.74 | 0.56 | 0.89 | 1.72 | 3.99 |
| AR | 0.76 | 1.01 | 1.70 | 3.05 | 0.47 | 0.70 | 1.15 | 1.81 | 0.46 | 0.72 | 1.15 | 1.92 |
| Pro-MSF | **0.70** | **0.94** | **1.65** | **2.89** | **0.42** | **0.62** | **1.02** | **1.72** | **0.40** | **0.62** | **1.10** | **1.76** |

visual results of the proposed method against 4 state-of-the-art methods: MLS (Bose & Ahuja 2006), JGF (Liu et al. 2013), TGV (Ferstl et al. 2013) and AR (Yang, Ye, Li, Hou & Wang 2014). Overall, the results of MLS and JGF still contain considerable amount of noise, which indicates the limited denoising ability of them. The depth maps upsampled by TGV, AR and the proposed method are cleaner. However, from the highlighted regions, texture-copying artifacts (e.g. the second row in Fig. 4.15) and blurring depth edges (e.g. the fifth row in Fig. 4.15) can be observed in the results of TGV and AR. Results of the proposed method do not have such texture-copying artifacts. In addition, the edge of the rectangular box in "Shark" dataset highlighted by red square (the fourth row in Fig. 4.15) is more sharp in the result of Pro-MSF, which shows that depth edges can be efficiently preserved by using the proposed method.

### Experiments for Depth Completion

The experiments for depth map completion are performed on the synthetic dataset (Yang, Ye, Li, Hou & Wang 2014) and the real dataset (Silberman et al. 2012) which are shown in two parts respectively.

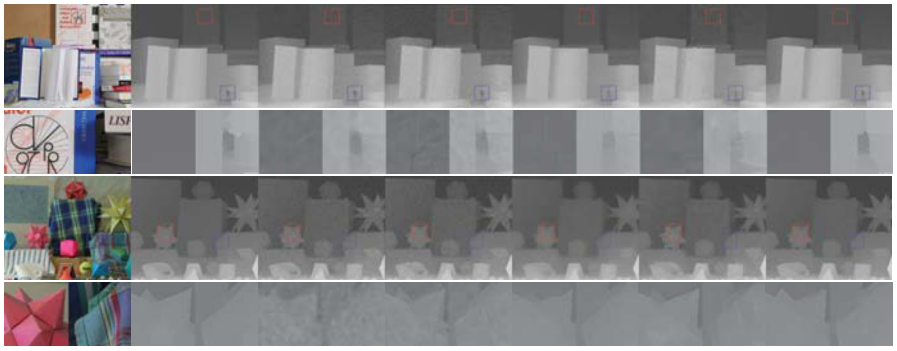
Table 4.14: Quantitative Upsampling Results (in MAD) on Noisy Middlebury Datasets "Reindeer", "Laundry" and "Dolls"

| Datasets / Methods | Reindeer | | | | Laundry | | | | Dolls | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 2× | 4× | 8× | 16× | 2× | 4× | 8× | 16× | 2× | 4× | 8× | 16× |
| Bicubic | 3.39 | 3.52 | 3.82 | 4.45 | 3.35 | 3.49 | 3.77 | 4.35 | 3.28 | 3.34 | 3.47 | 3.72 |
| MLS | 0.92 | 1.49 | 2.86 | 3.53 | 0.94 | 1.53 | 2.83 | 3.58 | 0.81 | 1.34 | 2.57 | 3.09 |
| Guided | 1.29 | 1.99 | 2.99 | 4.14 | 1.28 | 2.05 | 3.04 | 4.10 | 1.19 | 1.94 | 2.80 | 3.50 |
| NLMR | 1.20 | 1.60 | 2.40 | 3.97 | 1.28 | 1.63 | 2.20 | 3.34 | 1.14 | 1.54 | 2.07 | 3.02 |
| JGF | 2.18 | 2.40 | 2.89 | 3.94 | 2.16 | 2.37 | 2.85 | 3.90 | 2.09 | 2.22 | 2.49 | 3.25 |
| TGV | 0.59 | 0.84 | 1.75 | 4.40 | 0.61 | 1.59 | 1.89 | 4.16 | 0.66 | 1.63 | 1.75 | 3.71 |
| AR | **0.48** | 0.80 | **1.29** | **2.02** | **0.51** | 0.85 | 1.30 | 2.24 | 0.59 | 0.91 | 1.32 | 2.08 |
| Pro-MSF | 0.51 | **0.78** | 1.32 | 2.20 | **0.51** | **0.80** | **1.17** | **2.15** | **0.50** | **0.86** | **1.26** | **2.00** |

Table 4.15: Quantitative Upsampling Results (in MAD) on ToF-Mark Datasets

| Datasets | Bicubic | OMRF | Guided | MLS | JBU | JGF | NLMR | TGV | AR | Pro-Soft | Pro-MSF |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Books | 16.23 | 13.87 | 14.51 | 14.50 | 14.78 | 17.39 | 14.31 | 11.90 | 12.45 | 12.23 | **11.80** |
| Shark | 17.78 | 16.07 | 16.62 | 16.26 | 17.15 | 18.17 | 15.88 | 14.47 | 14.71 | 14.14 | **13.90** |
| Devil | 16.66 | 15.36 | 24.97 | 14.97 | 25.46 | 19.02 | 15.36 | 13.90 | 13.83 | 13.71 | **13.51** |

**Depth completion on synthetic datasets** The proposed method is evaluated for depth map completion on the datasets provided by (Yang, Ye, Li, Hou & Wang 2014) which manually adds some holes in the ground truth of Middlebury datasets (*Middlebury Datasets [Online]* n.d.). The holes consist of structural errors and random missing which are generated near depth edges and on smooth regions respectively. Tab. 4.16 shows results of the proposed method (Pro-MSF) compared with 6 benchmark and state-of-the-art methods: Bicubic, MLS (Bose & Ahuja 2006), JBF (Kopf et al. 2007), Guided (He et al. 2010), AR (Yang, Ye, Li, Hou & Wang 2014) and Pro-Soft. Results of the proposed method obtain the lowest MAD for all datasets, which proves its effectiveness. Fig. 4.16 shows results of the proposed method compared with that of MLS, Guided, JBF and Pro-Soft. From highlighted regions, some texture-copying artifacts are observed in the results of JBF. Guided



**(a)** **(b)** **(c)** **(d)** **(e)** **(f)** **(g)**

**Figure 4.14:** Visual comparison of upsampled (8×) depth maps on noisy Middlebury datasets "Book" and "Moebius", (a) HR color images, (b) ground truth depth maps, LR depth maps upsampled by: (c) Guided, (d) JGF, (e) TGV, (f) NLMR and (g) the proposed method.

suffers from blurring depth edges. On the contrary, the proposed method not only maintains more details (e.g. the region highlighted by red square in the fourth row) but also better preserves depth edges compared with MLS and Pro-Soft.

**Depth completion on real datasets** The proposed method is also evaluated for depth map completion on NYU datasets (Silberman et al. 2012) in which the depth maps are captured by Kinect. Fig. 4.17 shows results of the proposed method compared with 4 state-of-the-art methods: Colorization (Levin et al. 2004), JBF (Kopf et al. 2007), AR (Yang, Ye, Li, Hou & Wang 2014) and Pro-Soft.

It is shown that AR suffers from texture-copying artifacts. Colorition and JBF cannot maintain sharp straight depth edge (e.g. the region highlighted by blue square in the second row). Some blurring depth edges are shown in the results of AR and Pro-Soft (e.g. the region highlighted by red square in the second row). Compared with such methods, the proposed method can best preserve depth edges and significantly mitigate texture-copying artifacts.

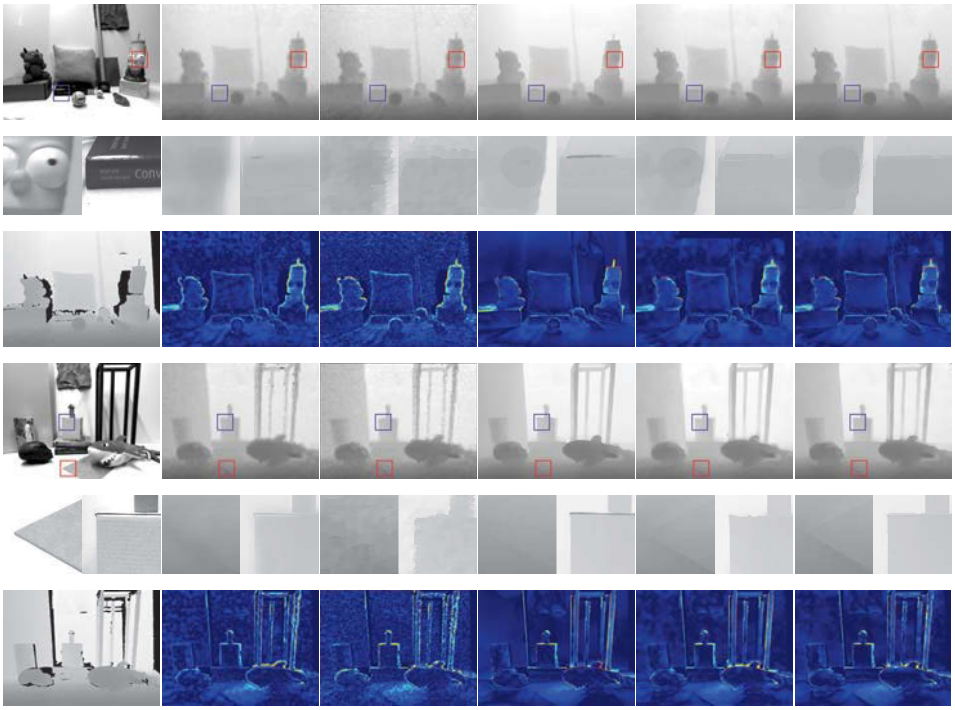



**Figure 4.15:** Visual comparison of upsampled depth maps on ToF-Mark datasets "Devil" and "Shark", (a) HR intensity images and ground truth depth maps, LR depth maps upsampled by: (b) MLS, (c) JGF, (d) TGV, (e) AR and (f) the proposed method.

### Running Time Comparison

The platform to carry out the experiments is a desktop with 3.33 GHz CPU and 12G RAM. The proposed method is implemented based on unoptimized Matlab and C++ code. This part compares the average running time of the proposed method with 3 optimization-based approaches (NLMR (Park et al. 2014), TGV (Ferstl et al. 2013) and AR (Yang, Ye, Li, Hou & Wang 2014)) which belong to the same category of approach as the proposed method. In the proposed method, the overall computation time mainly depends on the upsampling factor. Larger the upsampling factor is, more computation time

Table 4.16: Quantitative Completion Results (in MAD) on Synthetic Datasets

| Methods \ Datasets | Art | Book | Moebius | Reindeer | Laundry | Dolls |
|---|---|---|---|---|---|---|
| Bicubic | 0.90 | 0.61 | 0.66 | 0.95 | 0.91 | 0.76 |
| MLS | 0.91 | 0.58 | 0.72 | 0.68 | 0.72 | 0.82 |
| JBF | 0.84 | 0.63 | 0.69 | 0.92 | 0.88 | 0.76 |
| Guided | 1.20 | 0.63 | 0.67 | 0.96 | 0.94 | 0.76 |
| AR | 0.58 | 0.53 | 0.60 | 0.68 | 0.75 | 0.69 |
| Pro-Soft | 0.60 | 0.52 | 0.56 | 0.70 | 0.71 | 0.68 |
| Pro-MSF | **0.54** | **0.50** | **0.53** | **0.64** | **0.67** | **0.66** |

Table 4.17: Average Running Time Comparison (in Seconds) among Optimization-based Approaches

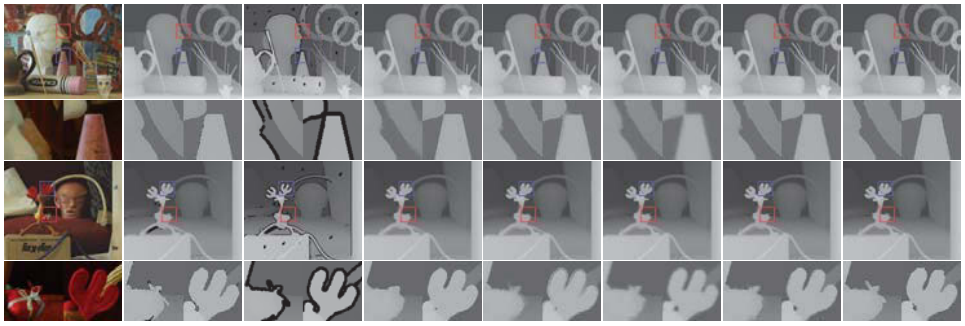
| Datasets \ Methods | NLMR | TGV | AR | Pro-MSF |
|---|---|---|---|---|
| Middlebury | 167.45 | 4073.63 | 280.51 | 2×: 173.11 4×: 197.30 8×: 228.86 16×: 264.08 |
| ToF-Mark | 55.79 | 1357.68 | 132.98 | 62.68 |

is consumed.

Tab. 4.17 lists the average running time in seconds of different approaches on Middlebury datasets and ToF-Mark datasets. TGV requires thousands of iterations to converge and takes 4000 seconds which is not suitable to a large-scale optimization problem. Overall, the proposed method provides improved performance at the cost of extra running time than NLMR. In addition, it is faster than AR and TGV.

### 4.5.4 Conclusion

This section proposes a novel guided depth map enhancement method via MRF optimization. The key contributions are two-folds: the first one is to compute the regularization affinities based on the proposed tree distance in the domain of Minimum Spanning Forest. Such structural scheme can better



**(a)** **(b)** **(c)** **(d)** **(e)** **(f)** **(g)** **(h)**

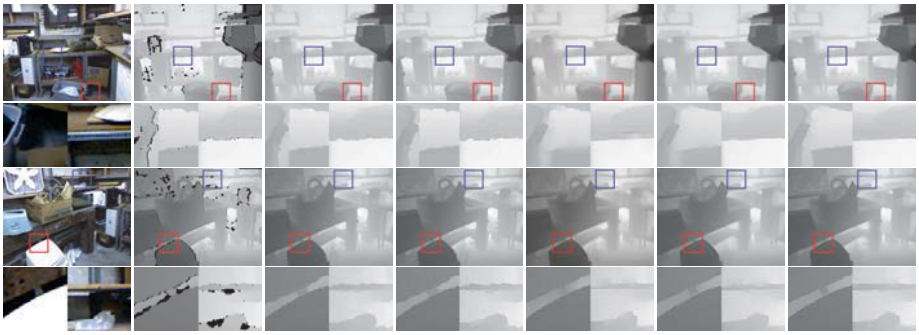
**Figure 4.16:** Visual comparison of enhanced depth maps on synthetic datasets, (a) color images, (b) ground truth depth maps, (c) low-quality depth maps with holes, depth maps enhanced by: (d) MLS, (e) JBF, (f) Guided, (g) Pro-Soft and (h) the proposed method.

preserve depth edges. The second one is to explicitly embed the edge inconsistency measurement model into the edge weights inside each Minimum Spanning Tree which is constructed within a super-pixel. It significantly mitigates texture-copying artifacts. In addition, the proposed method is further improved by double-threshold bandwidth adaption scheme which provides a prior whether the depth of pixel pairs in regularization term should be close.

The proposed method is evaluated on Middlebury, ToF-Mark and NYU datasets for depth map SR and depth map completion tasks. Compared with the state-of-the-art methods, all the experimental results demonstrate that the proposed method can significantly mitigate texture-copying artifacts and better preserve depth edges even when the quality of depth map is very low (e.g. the upsampling factor is large).

## 4.6 Summary

This chapter presents two measurement model for edge inconsistency between depth edge map and corresponding color edge map via hard-decision and soft-decision manners. Such models can adaptively control the guid-



**(a)** **(b)** **(c)** **(d)** **(e)** **(f)** **(g)**

**Figure 4.17:** Visual comparison of enhanced depth maps on NYU datasets, (a) color images, (b) low-quality depth maps with holes, depth maps enhanced by: (c) Colorition, (d) JBF, (e) AR, (f) Pro-Soft and (g) the proposed method.

ance from HR color image when embedding into MRF energy function. The soft-decision one can provide more robust performance than hard-decision counterpart due to fine-grained evaluation in a numerical way. In addition, minimum spanning forest (MSF) is proposed to extract depth structure. The best performance is provided by combining MSF and soft-decision edge inconsistency measurement model together which is validated by sufficient experimental results in Section 4.5.

# Chapter 5

# Guided Depth Map Super-resolution via Deep Learning

## 5.1 Introduction and Motivation

The depth information of a scene is essential in many applications such as autonomous navigation, 3D reconstruction, human-computer interaction and virtual reality. Consumer-level sensors provide a convenient way to obtaining depth of real scenes. However, by considering the cost of sensors, the resolution of sensed depth map are low which introduces the problem of depth map super-resolution (SR). Following machine learning widely used in color image SR, such tools (e.g., sparse coding and deep learning) are applied to single depth map SR (Ferstl et al. 2015, Xie, Chou, Feris & Sun 2014). Due to highly illness, single depth SR only works well when upsampling scale is small (2×, 4×). But depth maps sensed by Time-of-Flight (ToF) sensor are typically about 200×200. A larger upsampling scale (e.g. 8×) is needed to obtain high-resolution (HR) depth map.

As chapter 4 shown, guided depth map SR can get more satisfied results for 8× even 16× using the edge guidance from HR color image. Nevertheless

such model-based methods may not optimal in the specific RGB-D data and are prone to introduce artifacts caused by edge misalignment between depth map and guided color image. Therefore, a straight way is to learn the guidance from HR color image based on external datasets. Such learning-based methods do not explicitly formulate an optimization problem as the global optimization-based methods (Diebel & Thrun 2005, Lu et al. 2011, Park et al. 2014, Ferstl et al. 2013, Yang, Ye, Li, Hou & Wang 2014) or design a fixed filter as the filter-based methods (Kopf et al. 2007, Liu et al. 2013, He et al. 2010, Min et al. 2012, Hua et al. 2016). Some representative methods are proposed. As a pioneer work, Li et al. (Li et al. 2012) propose a guided depth SR method based on sparse coding. It jointly trains three dictionaries for registered patches of low-resolution (LR) depth maps, HR depth maps and HR color images. In the reconstruction phase, the depth maps are enhanced through sparse representation of learned dictionaries. Kwon et al. (Kwon et al. 2015) train three dictionaries of HR depth patches, LR depth patches and HR RGB patches via multi-scale learning scheme. In addition, this method optimizes an objective function which explicitly constrains consistent reconstruction between overlapping patches. Kiechle et al. (Kiechle et al. 2013) exploit the co-sparsity of analysis operators applied on RGB-D image pair, and reconstructed the HR depth map through data fidelity and color-guided sparsity constraint. In addition to sparse coding, recently, convolutional neural network (CNN) shows stronger representative ability than traditional methods (SVM, sparse coding). Dong et al. (Dong et al. 2016) firstly introduce CNN into color image SR via Fully Convolutional Network. Unlike Dong et al. (Dong et al. 2016), the proposed method learns the upsampling kernels instead of fixed ones (e.g., Bicubic upsampling kernel). Then Hui et al. (Hui et al. 2016) firstly propose a Multi-Scale Guided convolutional network for depth map SR. It complements LR depth features with HR intensity features through a multi-scale fusion strategy which progressively resolves ambiguity in depth map SR using the HR intensity features at different levels. Although Hui et al. (Hui et al. 2016) can provide good depth

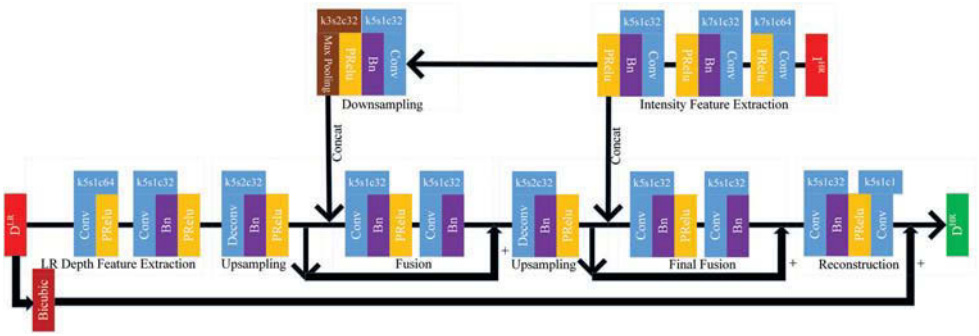



**Figure 5.1:** The architecture of MFR-SR for the case 4×. The networks for other upsampling scale are in the same manner as 4×

SR results, it is not an end-to-end model which has a preprocessing to extract low frequency component by fixed low-pass filter. The input of the network is the high frequency component of input RGB-D images by subtracting the low frequency counterpart above. Such low frequency component is added to the output of the network to reconstruct HR depth map. However, such hand-craft preprocessing is not robust for noisy depth SR since noise is high frequency as well and is not relevant to the guided image. Experimental section shows its poor performance on noisy depth SR. To overcome the drawback above, this chapter adopts residual learning to replace such hand-craft preprocessing and proposes an end-to-end residual network. Generally, it upsamples LR depth map via multi-scale reconstruction. Furthermore, the residual learning is introduced in each scale-dependent reconstruction sub-network. This coarse-to-fine scheme can reconstruct high-resolution depth via multi-frequency synthesis which in turn makes the network to better restore structure in the certain scale. In addition, to improve the robustness of training, batch normalization is applied in the proposed network. More details will be addressed in the next section. In the rest of this chapter, the proposed Multi-scale Fusion Residual network for depth map SR is called as MFR-SR.

## 5.2 MFR-SR Construction

Following the concept of guided depth SR, MFR-SR is to train a generating function $G\left(\mathbf{D^{LR}}, \mathbf{I^{HR}}, \boldsymbol{\phi}\right)$ instead of hand-craft model that restores HR depth $\mathbf{D^{HR}}$ for LR depth input $\mathbf{D^{LR}}$ under certain guidance from a corresponding HR color image $\mathbf{I^{HR}}$. And $\boldsymbol{\phi}$ is the trainable parameters.

Fig. 5.1 is the architecture of MFR-SR for the case of $4\times$, where $k, s$ and $c$ stand for kernel size of filter, stride and output channel amount respectively. Overall, MFR-SR consists of depth/intensity feature extraction units, depth upsampling/intensity downsampling unit, fusion unit and reconstruction unit. The roles of main units in the network are briefly explained as below;

Feature Extraction: It is observed that Bicubic interpolation can provide satisfied SR results for smooth regions, but it fails to recover sharp edges. According to this observation, each spectral components of an image need a special upsampling strategy. Therefore, in MSR-SR, the LR depth map and HR intensity image are firstly decomposed into multiple spectral components using different trainable convolutional kernels which leads to spectral-dependent mapping function learning. To better extract features of depth/intensity, MFR-SR uses more blocks in feature extraction unit than Hui et al. (Hui et al. 2016).

Fusion: This type of unit generates high quality depth spectral components for multiple scales. The corresponding intensity spectral components are used as guidance to focus on high frequency detail restoration for the low quality depth spectral components which is upsampled by itself. Therefore, such unit is the core of network which performs intensity guidance learning for depth SR.

In MSR-SR, $\mathbf{D^{LR}}$ is progressively upsampled ($2\times$) in depth upsampling units to obtain $\mathbf{D^{HR}}$. Meanwhile, $\mathbf{I^{HR}}$ is progressively downsampled to provide multiple-level guidance for fusion units. The input of each fusion unit are upsampled depth feature maps and corresponding guided intensity feature maps which are concatenated with each other. Such feature maps can be

explained as spectral components of images. MFR-SR borrows the concept of residual learning (He, Zhang, Ren & Sun 2015), which uses the bicubic interpolated depth map as base layer and train the residual between the base layer and the ground truth depth map. In addition, based on the stronger relation between high frequency of depth spectral components and corresponding intensity parts, during each fusion stage, a skip connection is designed to pass the sub-base layer generated by single depth spectral components upsampling to the end of this fusion unit. Such sub-base layer is the low frequency part of depth spectral components in current scale (resolution) which is similar to the meaning of base-layer. Therefore, the guidance from intensity spectral components in the corresponding scale focus on restoring relative high frequency detail to complement the sub-base layer. Such multi-scale residual learning leads to the coarse-to-fine depth reconstruction which is more efficient. Compared with the propose method, the high-frequency domain training used in Hui et al. (Hui et al. 2016) is less efficient for noisy depth SR since noise is high frequency as well and is not relevant to the guided image. This view is supported in the experimental results.

In addition, compared with Hui et al. (Hui et al. 2016) which does not use batch-normalization layer, the core block in MFR-SR consists of a convolutional layer or transposed convolutional followed by a batch-normalization layer (Bn) (Ioffe & Szegedy 2015) and a ParametricReLU layer (Prelu) (He et al. 2015) as the activation function which improves the robustness of training. Batch-normalization layers are skipped in the first and the last blocks. ParametricReLU is skipped in the last block. The blocks of the specific setting can be expressed as below;

$$\mathbf{F^O} = \delta\left(\mathbf{W} * \mathbf{F^I} + \mathbf{b}\right) \tag{5.1}$$

$$\mathbf{F^O} = \delta\left(Bn\left(\mathbf{W} * \mathbf{F^I}\right)\right) \tag{5.2}$$

$$\mathbf{F^O} = \delta\left(Bn\left(\mathbf{W} \star \mathbf{F^I}\right)\right) \tag{5.3}$$

$$\delta(x) = \max(0, x) + \alpha \min(0, x) \tag{5.4}$$

where $\mathbf{F^I}$ and $\mathbf{F^O}$ are input and output of the block. $*$ and $\star$ are stands for convolution and transposed convolution respectively. $\delta$ and $Bn$ are Prelu with a learning parameter $\alpha$ and batch-normalization function. $\mathbf{W}$ and $\mathbf{b}$ represent kernel of filter and bias in convolutional layer respectively. Since bias parameter are introduced in batch-normalization, there is no bias $\mathbf{b}$ in Eq. (5.2) and Eq. (5.3).

By defining $\boldsymbol{\phi}$ as the learning parameter set controlling the forward process, the loss function $L$ of MFR-SR is the mean squared error (MSE) on all the $M$ training images which is expressed as below:

$$L(\boldsymbol{\phi}) = \frac{1}{M} \left\| G\left(\mathbf{D^{LR}}, \mathbf{I^{HR}}, \boldsymbol{\phi}\right) - \mathbf{D^{HR}} \right\|_F^2 \tag{5.5}$$

The loss is minimized by Adam optimizer (Kingma & Ba 2014) with $\beta_1 = 0.9$.

## 5.3 Experiments

Firstly, this section introduces the detail of training data generation and model training. Then experimental results are shown which validates the effectiveness of the propose method.

### 5.3.1 Training Data

The original training dataset consisting of 5000 RGB-D image pairs with resolution 512×512 is provided by (Riegler et al. 2016) which is generated using the open source Mitsuba Render Software (*Mitsuba Renderer [Online]* n.d.). This automatic dataset generation is scripted by randomly placing different objects (cubes, spheres and planes) in varying dimensions in the scene. In addition, the objects are randomly textured using samples from the publicly available Describable Textures Dataset (Cimpoi, Maji, Kokkinos, Mohamed & Vedaldi 2014). The light intensity and position is also slightly varied during the data generation. In this experiment, the intensity is used for guided

image, therefore, the RGB color images are transferred to intensity image beforehand. The amount of RGB-D pairs used for training and validation are 4300 and 700 respectively. Furthermore, due to limitation of resource and rate of convergence, the images cannot be trained in full-size. The image pairs should be firstly cut into sub-images. Such sub-image pairs (i.e., HR intensity image and HR depth map) are 128×128 with 64 pixels overlap. The corresponding LR depth map is downsampled via bicubic which are 64×64, 32×32, 16×16 and 8×8 for 2×, 4×, 8× and 16× respectively. To simulate depth maps obtained in real situation, a Gaussian noise with variance 25 and mean 0 is added to LR depth maps. Lastly, triple training pairs ($\mathbf{D^{HR}}$, $\mathbf{I^{HR}}$ and $\mathbf{D^{LR}}$ ) are scaled into [0, 1].

### 5.3.2 Training Detail

The network is built on the TensorFlow platform (*tensorflow [Online]* n.d.) and trained on NVIDIA Titan 1080Ti GPU. Note that the size of input images can be arbitrary size as it is fully convolutional. The batch size is 32 and the network is trained at learning rate 1e-4 for 10 epochs followed by another 10 epochs at learning rate 1e-5. During test phase, MFR-SR turns batch-normalization update off to obtain an output that deterministically depends only on the input. The proposed method trains a specific network for each upscaling factor (e.g., 2×, 4×, 8×, 16×) respectively.

### 5.3.3 Experimental Results

The network is tested on 10 hole-filled Middlebury RGB-D datasets ("Art", "Books", "Moebius", "Laundry", "Dolls", "Reindeer", "Cones", "Teddy", "Tsukuba" and "Venus") which are used in many literatures. Such testing image pairs are classified into $A$ (including the first 6 ones) and $B$ (including the last 4 ones) sets based on the original HR resolution. The ground truth depth maps are firstly downsampled by bicubic interpolation at multiple scales (e.g., 2×, 4×, 8× and 16× for $A$, 2×, 4× and 8× for $B$). Then, a

Table 5.1: Quantitative SR Results (in RMSE) on Noisy Middlebury Datasets "Art", "Book" and "Moebius"

| Methods \ Datasets | Art | | | | Book | | | | Moebius | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 2× | 4× | 8× | 16× | 2× | 4× | 8× | 16× | 2× | 4× | 8× | 16× |
| Bicubic | 4.78 | 5.54 | 6.74 | 9.04 | 4.20 | 4.39 | 4.68 | 5.30 | 4.16 | 4.31 | 4.54 | 5.02 |
| MLS | 4.12 | 4.45 | 6.25 | 8.54 | 1.90 | 2.47 | 4.05 | 4.85 | 1.82 | 2.27 | 3.76 | 4.52 |
| Guided | 3.85 | 4.24 | 5.35 | 8.08 | 1.84 | 2.21 | 3.00 | 4.41 | 1.81 | 2.63 | 3.68 | 4.71 |
| NLMR | 4.65 | 5.99 | 8.01 | 11.01 | 2.16 | 2.72 | 3.29 | 4.91 | 2.06 | 2.50 | 3.27 | 4.61 |
| JGF | 4.17 | 5.26 | 7.12 | 10.27 | 2.95 | 3.41 | 4.10 | 5.48 | 2.84 | 3.20 | 3.92 | 5.24 |
| TGV | 3.08 | 3.94 | 7.26 | 12.05 | 1.50 | 2.14 | 2.88 | 4.73 | 1.46 | 1.98 | 3.01 | 6.11 |
| AR | 3.19 | 3.46 | 4.91 | <u>7.46</u> | 1.32 | 1.99 | 2.77 | <u>3.59</u> | 1.31 | **1.66** | <u>2.52</u> | <u>3.51</u> |
| Pro-MSF | 3.33 | 3.55 | 5.02 | **7.14** | 1.46 | <u>1.88</u> | **2.58** | 3.66 | 1.57 | <u>1.75</u> | **2.34** | **3.48** |
| ECCV-CNN | <u>2.06</u> | <u>3.38</u> | <u>4.86</u> | 8.12 | <u>1.27</u> | 1.99 | 2.96 | 4.39 | <u>1.29</u> | 1.84 | 2.66 | 4.46 |
| MFR-SR | **2.03** | **3.28** | **4.75** | **7.14** | **1.11** | **1.84** | <u>2.60</u> | **3.51** | **1.18** | 1.95 | 2.79 | 3.73 |

Gaussian noise is added to the downsampled depth maps to generate LR depth maps. This subsection provides both quantitative and qualitative evaluations on guided depth SR. The proposed MFR-SR is compared with 10 benchmark and state-of-the-art methods which are Bicubic interpolation (bicubic), Spatial-depth super resolution for range images (JBUV) (Yang et al. 2007), Guided image filtering (Guided) (He et al. 2010), Weighted mode filter (WMF) (Min et al. 2012), Edge-weighted NLM-regularization (NLMR) (Park et al. 2014), Joint geodesic filtering (JGF) (Liu et al. 2013), Total generalized variation (TGV) (Ferstl et al. 2013), Moving least squares filter (MLS) (Bose & Ahuja 2006), CNN-based guided depth SR (ECCV-CNN) (Hui et al. 2016), Auto-regression model (AR) (Yang, Ye, Li, Hou & Wang 2014) and Pro-MSF proposed in Section 4.5. ECCV-CNN is retrained using the same training data as MFR-SR. Quantitative measure metric is root mean squared error (RMSE). The best RMSE for each evaluation is in bold, whereas the second best one is underlined.

Table 5.2: Quantitative SR Results (in RMSE) on Noisy Middlebury Datasets "Reindeer", "Laundry" and "Dolls"

| Datasets / Methods | Reindeer | | | | Laundry | | | | Dolls | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 2× | 4× | 8× | 16× | 2× | 4× | 8× | 16× | 2× | 4× | 8× | 16× |
| Bicubic | 4.51 | 4.95 | 5.71 | 7.12 | 4.37 | 4.74 | 5.35 | 6.53 | 4.17 | 4.30 | 4.51 | 4.90 |
| MLS | 2.82 | 3.41 | 5.22 | 6.29 | 2.71 | 3.21 | 4.75 | 6.19 | 1.63 | 2.14 | 3.55 | 4.29 |
| Guided | 2.63 | 3.44 | 4.80 | 6.73 | 2.33 | 3.23 | 4.52 | 6.22 | 1.80 | 2.65 | 3.71 | 4.62 |
| NLMR | 3.11 | 3.86 | 5.33 | 7.56 | 2.99 | 3.63 | 4.51 | 6.35 | 2.07 | 2.61 | 3.33 | 4.45 |
| JGF | 3.47 | 4.27 | 5.61 | 7.52 | 3.23 | 3.95 | 5.12 | 7.20 | 2.81 | 3.11 | 3.61 | 4.80 |
| TGV | 3.08 | 4.20 | 4.65 | 9.03 | 2.62 | 5.05 | 4.45 | 8.06 | 1.49 | 2.86 | 2.82 | 5.41 |
| AR | 2.03 | 2.72 | 3.81 | 4.93 | 1.74 | 2.79 | **3.24** | 5.22 | 1.42 | **1.70** | 2.44 | **3.21** |
| Pro-MSF | 2.53 | 3.00 | 3.94 | **5.26** | 1.94 | 2.59 | 3.51 | **5.01** | 1.40 | 1.84 | **2.20** | 3.42 |
| ECCV-CNN | 1.58 | 2.53 | 3.93 | 6.15 | **1.52** | 2.49 | 3.65 | 6.83 | 1.38 | 1.95 | 2.73 | 4.26 |
| MFR-SR | **1.53** | **2.45** | **3.78** | 5.46 | 1.66 | **2.46** | 3.40 | 5.22 | **1.30** | 1.92 | 2.65 | 3.56 |

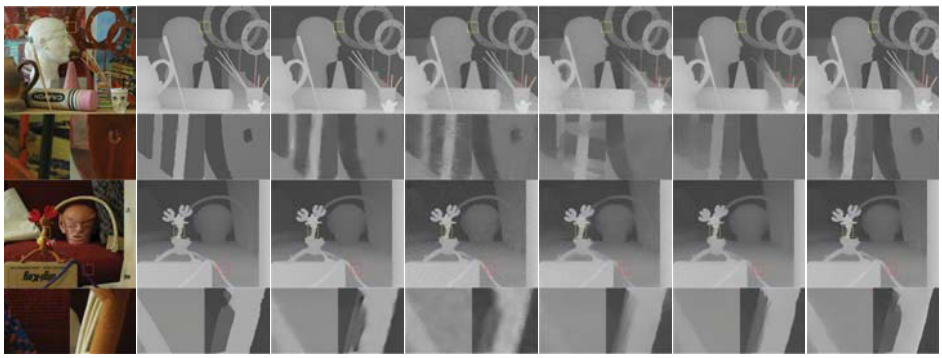
### Quantitative Evaluation

The HR resolution of datasets $A$ is 1376×1088. Tab. 5.1 and Tab. 5.2 show the RMSE results of dataset A under four upsampling scales (i.e., 2×, 4×, 8× and 16×). From such tables, it is shown that the proposed method obtains the lowest or the second lowest RMSE for most cases. The denoising ability of JGF (Liu et al. 2013) is very poor. The performances of NLMR (Park et al. 2014), MLS (Bose & Ahuja 2006) and Guided (He et al. 2010) are similar which are worse than the propose method. TGV (Ferstl et al. 2013) can obtain good results when upsampling factor is low (i.e., 2×, 4×), but it lacks robustness in the cases of large upsampling factors (i.e., 8×, 16×). AR (Yang, Ye, Li, Hou & Wang 2014) and Pro-MSF proposed in Section 4.5 are more robust than the method above. However, they show inferior performance than the proposed network. ECCV-CNN is not robust for large upsampling scale (e.g., 16×), since training in high-frequency domain is less efficient when the LR depth is noisy. Such training strategy leads to worse performance for denoising under large upsampling scale.

The resolution of HR depth map in dataset $B$ is about 475×350. There-

Table 5.3: Quantitative SR Results (in RMSE) on Noisy Middlebury Datasets "Cones", "Teddy", "Tsukuba" and "Venus"

| Methods \ Datasets | Cones | | | Teddy | | | Tsukuba | | | Venus | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 2× | 4× | 8× | 2× | 4× | 8× | 2× | 4× | 8× | 2× | 4× | 8× |
| Bicubic | 4.81 | 5.60 | 6.88 | 4.51 | 5.02 | 5.70 | 7.07 | 9.53 | 13.06 | 4.25 | 4.52 | 4.91 |
| JBUV | 3.20 | 4.31 | 6.21 | 2.52 | 3.48 | 4.88 | 6.78 | 8.22 | 12.14 | 1.83 | 2.77 | 4.03 |
| NLMR | 3.69 | 5.06 | 7.42 | 3.14 | 3.98 | 5.93 | 7.97 | 10.19 | 14.65 | 2.39 | 2.86 | 3.80 |
| WMF | 3.29 | 4.43 | 6.24 | 2.65 | 3.57 | 4.88 | 5.98 | 8.04 | **12.02** | 2.01 | 2.83 | 3.97 |
| TGV | 3.36 | 4.31 | 7.74 | 2.67 | 3.20 | 4.93 | 7.20 | 10.10 | 16.08 | 1.58 | **1.91** | **2.65** |
| ECCV-CNN | 2.55 | 4.41 | 6.16 | 2.31 | 3.41 | 5.10 | **4.60** | 9.14 | 12.89 | 1.42 | 2.67 | 4.15 |
| MFR-SR | **2.44** | **3.85** | **5.74** | **2.20** | **3.11** | **4.61** | 5.20 | **7.85** | 12.84 | **1.31** | 2.45 | 3.16 |



**(a)** **(b)** **(c)** **(d)** **(e)** **(f)** **(g)**

**Figure 5.2:** The visual quality comparison for depth map SR with noise on "Art" and "Reindeer" datasets. (a) HR color images, (b) ground truth, depth maps are upsampled (8×) by (c) AR, (d) Guided, (e) NLMR, (f) TGV, (g) the proposed method.

fore, the SR experiments are conducted on only three scales (2×, 4× and 8×). Tab. 5.3 lists the results comparison in terms of RMSE. The proposed method reaches lowest RMSE in most cases. However, since the simple structure of "Venus", the advantage of the proposed method is not shown on this dataset.

**Qualitative Evaluation**

Fig. 5.2 illustrates the SR results of the proposed method compared with state-of-the-art ones: AR (Yang, Ye, Li, Hou & Wang 2014), Guided (He et al. 2010), NLMR (Park et al. 2014) and TGV (Ferstl et al. 2013). From the highlighted regions, it is shown that the proposed network can generate clear depth maps with sharper edges. On the contrary, there are noticeable noise left in the result of Guided. AR and NLMR suffer from over-smooth artifacts. TGV fail recovering image structure (e.g., the result on "Art" dataset).

## 5.4 Summary

This chapter proposes a CNN-based guided depth SR method. Such end-to-end neural network progressively upsamples LR depth map to its HR counterpart under the guidance of corresponding intensity images. It borrows the concept of residual learning and batch normalization which are beneficial to coarse-to-fine reconstruction and training. The experimental results on 10 noisy Middlebury RGB-D image pairs are shown the improvement of the proposed method compared with state-of-the-art methods.

# Chapter 6

# Conclusions and Future Work

## 6.1 Conclusions

This thesis presents several techniques to address the problem in high-quality depth map/sequence acquisition.

Chapter 3 firstly proposes a passive depth-acquisition method via fast stereo matching, which takes the adaptive matching scheme and affine invariant feature into account. The adaptive matching scheme can significantly reduce computational complexity. And by introducing the invariant feature, more robust depth for textureless regions are estimated. Subsequently, a temporal consistency enhancement method is proposed for estimated multi-view depth sequences to mitigate flashing artifacts in rendered color images. This method explicitly considers reliability of depth and moving attribute of regions to mitigate error propagation among frames. Such two methods are validated by sufficient experiments.

In Chapter 4, three methods are proposed which improve performance progressively. Firstly, it proposes a depth super-resolution (SR) method based on hard-decision edge inconsistency measurement. Although it provides satisfied results for small upsampling factors (i.e., 2× and 4×), the performance is not robust for large upsampling factors. Then, a soft-decision edge inconsistency measurement is proposed to address the disadvantage of

its hard-decision counterpart. It uses a more fine-grained model which considers local and global structure. In the following step, such quantitative measurement is embedded into MRF-based model. It controls the efforts of the guidance from the color image. Therefore, the modified MRF can better mitigate texture-copying artifacts and preserve depth edges than embedding the hard-decision edge inconsistency measurement. Although promising depth enhancement results can be obtained by using the second method, there are some failed cases when the quality of the original depth map is too low due to non-structural affinity computation in MRF energy function. To overcome this drawback, it is further proposed to compute the regularization affinities based on tree distance in the domain of Minimum Spanning Forest. The soft-decision edge inconsistency measurement is embedded into the edge weights of each Minimum Spanning Tree which is constructed within a superpixel. In addition, such method is further improved by a double-threshold bandwidth adaption scheme which provides a prior regarding whether the depth of pixel pairs in the regularization term should be close. The sufficient experiments show the progressive improvement in SR performance for such three proposed methods.

Chapter 5 presents a CNN-based guided depth SR method. Such end-to-end neural network progressively upsamples LR depth map to its HR counterpart under the guidance of the corresponding intensity images. It borrows the concepts of residual learning and batch normalization which are beneficial to coarse-to-fine reconstruction and training. The experimental results on 10 noisy Middlebury RGB-D image pairs show the improvement of the proposed method compared with state-of-the-art methods.

Chapter 3 and Chapter 4 of this thesis are supported by the published conference and journal papers listed in **List of Publications**. Also, a conference paper is planned based on Chapter 5.

## 6.2 Future Work

Very recently, generative adversarial network has been introduced into more and more applications including object detection (Li, Liang, Wei, Xu, Feng & Yan 2017) and color image super-resolution (Ledig, Theis, Huszár, Caballero, Cunningham, Acosta, Aitken, Tejani, Totz, Wang et al. 2016). However, such technique has not been introduced into depth SR, especially for guided depth SR. For this research direction, some preliminary experiments have been conducted which uses the network proposed in Chapter 5 as the generative network. The principle in Wasserstein GAN (WGAN) (Arjovsky, Chintala & Bottou 2017) is adopted to construct the discriminator. Based on the preliminary results, it is found that the loss evaluated in image domain may be necessary in the task of SR, since the accuracy of depth map significantly affects the performance of real scene reconstruction. But such image domain loss may lead to smooth edges, which is not the expected goal for introducing generative adversarial network. In addition, the denoising ability is not satisfied when a complex loss consisting of image domain loss and WGAN loss is adopted. For future work, some insights are expected to address such problems.

# Bibliography

Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P. & Süsstrunk, S. (2012), 'Slic superpixels compared to state-of-the-art superpixel methods', *IEEE transactions on pattern analysis and machine intelligence* **34**(11), 2274–2282.

Arjovsky, M., Chintala, S. & Bottou, L. (2017), 'Wasserstein gan'.

Bao, L., Song, Y., Yang, Q., Yuan, H. & Wang, G. (2014), 'Tree filtering: Efficient structure-preserving smoothing with a minimum spanning tree', *IEEE Transactions on Image Processing* **23**(2), 555–569.

Bender, M. A. & Farach-Colton, M. (2000), The lca problem revisited, *in* 'Latin American Symposium on Theoretical Informatics', Springer, pp. 88–94.

Bobick, A. F. & Intille, S. S. (1999), 'Large occlusion stereo', *International Journal of Computer Vision* **33**(3), 181–200.

Bose, N. K. & Ahuja, N. A. (2006), 'Superresolution and noise filtering using moving least squares', *IEEE Transactions on Image Processing* **15**(8), 2239–2248.

Boykov, Y. & Kolmogorov, V. (2004), 'An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision', *IEEE transactions on pattern analysis and machine intelligence* **26**(9), 1124–1137.

Boykov, Y., Veksler, O. & Zabih, R. (2001), 'Fast approximate energy minimization via graph cuts', *IEEE Transactions on pattern analysis and machine intelligence* **23**(11), 1222–1239.

Cai, J. (2012), 'Integration of optical flow and dynamic programming for stereo matching', *IET image processing* **6**(3), 205–212.

Canny, J. (1986), 'A computational approach to edge detection', *IEEE Transactions on pattern analysis and machine intelligence* (6), 679–698.

Cho, J.-H., Kim, S.-Y., Ho, Y.-S. & Lee, K. H. (2008), 'Dynamic 3d human actor generation method using a time-of-flight depth camera', *IEEE Transactions on Consumer Electronics* **54**(4), 1514–1521.

Choi, O. & Jung, S.-W. (2014), 'A consensus-driven approach for structure and texture aware depth map upsampling', *IEEE Transactions on Image Processing* **23**(8), 3321–3335.

Choi, S., Kim, T. & Yu, W. (1997), 'Performance evaluation of ransac family', *Journal of Computer Vision* **24**(3), 271–300.

Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S. & Vedaldi, A. (2014), Describing textures in the wild, *in* 'Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition', pp. 3606–3613.

Comaniciu, D. & Meer, P. (2002), 'Mean shift: A robust approach toward feature space analysis', *IEEE Transactions on pattern analysis and machine intelligence* **24**(5), 603–619.

Cormen, T. H., Leiserson, C. E., Rivest, R. L. & Stein, C. (2001), *Introduction to algorithms*, Vol. 6, MIT press Cambridge, chapter Finding the closest pair of points.

Diebel, J. & Thrun, S. (2005), An application of markov random fields to range sensing, *in* 'NIPS', Vol. 5, pp. 291–298.

Dong, C., Loy, C. C., He, K. & Tang, X. (2016), 'Image super-resolution using deep convolutional networks', *IEEE transactions on pattern analysis and machine intelligence* **38**(2), 295–307.

Ferstl, D., Reinbacher, C., Ranftl, R., Rüther, M. & Bischof, H. (2013), Image guided depth upsampling using anisotropic total generalized variation, *in* 'Proceedings of the IEEE International Conference on Computer Vision', pp. 993–1000.

Ferstl, D., Ruther, M. & Bischof, H. (2015), Variational depth superresolution using example-based edge representations, *in* 'Proceedings of the IEEE International Conference on Computer Vision', pp. 513–521.

Ford Jr, L. & Fulkerson, D. (2009), Maximal flow through a network, *in* 'Classic papers in combinatorics', Springer, pp. 243–248.

Freedman, G. & Fattal, R. (2011), 'Image and video upscaling from local self-examples', *ACM Transactions on Graphics (TOG)* **30**(2), 12.

Fu, D., Zhao, Y. & Yu, L. (2010), Temporal consistency enhancement on depth sequences, *in* 'Picture Coding Symposium (PCS), 2010', IEEE, pp. 342–345.

Furukawa, Y., Curless, B., Seitz, S. M. & Szeliski, R. (2009), Manhattan-world stereo, *in* 'Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on', IEEE, pp. 1422–1429.

Gong, M. & Yang, Y.-H. (2007), 'Real-time stereo matching using orthogonal reliability-based dynamic programming', *IEEE Transactions on Image Processing* **16**(3), 879–884.

Goodfellow, I., Bengio, Y. & Courville, A. (2016), *Deep Learning*, MIT Press. `http://www.deeplearningbook.org`.

Hammersley, J. M. & Clifford, P. E. (1971), 'Markov random fields on finite graphs and lattices', Unpublished manuscript .

Hartley, R. & Zisserman, A. (2003), *Multiple view geometry in computer vision*, Cambridge university press.

Hawe, S., Kleinsteuber, M. & Diepold, K. (2013), 'Analysis operator learning and its application to image reconstruction', *IEEE Transactions on Image Processing* **22**(6), 2138–2150.

He, K., Sun, J. & Tang, X. (2010), Guided image filtering, *in* 'European conference on computer vision', Springer, pp. 1–14.

He, K., Zhang, X., Ren, S. & Sun, J. (2015), Delving deep into rectifiers: Surpassing human-level performance on imagenet classification, *in* 'Proceedings of the IEEE international conference on computer vision', pp. 1026–1034.

Hosni, A., Bleyer, M., Gelautz, M. & Rhemann, C. (2009), Local stereo matching using geodesic support weights, *in* 'Image Processing (ICIP), 2009 16th IEEE ternational Conference on', IEEE, pp. 2093–2096.

Hosni, A., Bleyer, M., Rhemann, C., Gelautz, M. & Rother, C. (2011), Real-time local stereo matching using guided image filtering, *in* 'Multimedia and Expo (ICME), 2011 IEEE International Conference on', IEEE, pp. 1–6.

Hua, K.-L., Lo, K.-H. & Wang, Y.-C. F. F. (2016), 'Extended guided filtering for depth map upsampling', *IEEE MultiMedia* **23**(2), 72–83.

Huber, P. J. (1973), 'Robust regression: asymptotics, conjectures and monte carlo', *The Annals of Statistics* pp. 799–821.

Hui, T.-W., Loy, C. C. & Tang, X. (2016), Depth map super-resolution by deep multi-scale guidance, *in* 'European Conference on Computer Vision', Springer, pp. 353–369.

Ioffe, S. & Szegedy, C. (2015), Batch normalization: Accelerating deep network training by reducing internal covariate shift, *in* 'International Conference on Machine Learning', pp. 448–456.

Jang, W.-D. & Kim, C.-S. (2012), Seqm: Edge quality assessment based on structural pixel matching, *in* 'Visual Communications and Image Processing (VCIP), 2012 IEEE', IEEE, pp. 1–6.

Ju, K., Wang, B. & Xiong, H. (2015), Structure-aware priority belief propagation for depth estimation, *in* 'Visual Communications and Image Processing (VCIP), 2015', IEEE, pp. 1–4.

Kiechle, M., Hawe, S. & Kleinsteuber, M. (2013), A joint intensity and depth co-sparse analysis model for depth map super-resolution, *in* 'Proceedings of the IEEE International Conference on Computer Vision', pp. 1545–1552.

Kim, J., Kwon Lee, J. & Mu Lee, K. (2016), Accurate image super-resolution using very deep convolutional networks, *in* 'The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)'.

Kingma, D. & Ba, J. (2014), 'Adam: A method for stochastic optimization', *arXiv preprint arXiv:1412.6980* .

Kolb, A., Barth, E., Koch, R. & Larsen, R. (2010), Time-of-flight cameras in computer graphics, *in* 'Computer Graphics Forum', Vol. 29, Wiley Online Library, pp. 141–159.

Kopf, J., Cohen, M. F., Lischinski, D. & Uyttendaele, M. (2007), Joint bilateral upsampling, *in* 'ACM Transactions on Graphics (TOG)', Vol. 26, ACM, p. 96.

Kuhn, H. W. (1955), 'The hungarian method for the assignment problem', *Naval research logistics quarterly* **2**(1-2), 83–97.

Kwon, H., Tai, Y.-W. & Lin, S. (2015), Data-driven depth map refinement via multi-scale sparse representation, *in* 'Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition', pp. 159–167.

Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z. et al. (2016), 'Photo-realistic single image super-resolution using a generative adversarial network', *arXiv preprint arXiv:1609.04802* .

Lei, C., Selzer, J. & Yang, Y.-H. (2006), Region-tree based stereo using dynamic programming optimization, *in* 'Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on', Vol. 2, IEEE, pp. 2378–2385.

Levin, A., Lischinski, D. & Weiss, Y. (2004), Colorization using optimization, *in* 'ACM transactions on graphics (tog)', Vol. 23, ACM, pp. 689–694.

Li, J., Liang, X., Wei, Y., Xu, T., Feng, J. & Yan, S. (2017), 'Perceptual generative adversarial networks for small object detection', *arXiv preprint arXiv:1706.05274* .

Li, Y., Xue, T., Sun, L. & Liu, J. (2012), Joint example-based depth map super-resolution, *in* '2012 IEEE International Conference on Multimedia and Expo', IEEE, pp. 152–157.

Liao, M. Z., Wei, L. & Chen, W. (2007), A novel affine invariant feature extraction for optical recognition, *in* 'Machine Learning and Cybernetics, 2007 International Conference on', Vol. 3, IEEE, pp. 1769–1773.

Liu, M.-Y., Tuzel, O. & Taguchi, Y. (2013), Joint geodesic upsampling of depth images, *in* 'Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition', pp. 169–176.

Liu, W., Chen, X., Yang, J. & Wu, Q. (2017), 'Robust color guided depth map restoration', *IEEE Transactions on Image Processing* **26**(1), 315–327.

Lo, K.-H., Wang, Y.-C. F. & Hua, K.-L. (2013), Joint trilateral filtering for depth map super-resolution, *in* 'Visual Communications and Image Processing (VCIP), 2013', IEEE, pp. 1–6.

Lu, J., Min, D., Pahwa, R. S. & Do, M. N. (2011), A revisit to mrf-based depth map super-resolution and enhancement, *in* '2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)', IEEE, pp. 985–988.

*Middlebury Datasets [Online]* (n.d.), `http://vision.middlebury.edu/stereo/data/`.

Min, D., Lu, J. & Do, M. N. (2012), 'Depth video enhancement based on weighted mode filtering', *IEEE Transactions on Image Processing* **21**(3), 1176–1190.

*Mitsuba Renderer [Online]* (n.d.), `http://www.mitsuba-renderer.org`.

Olgierd Stankiewicz, Krzysztof Wegner, M. W. (2009), A soft segmentation matching in depth estimation reference software (ders) 5.0, *in* 'ISO/IEC JTC1/SC29/WG11 MPEG 2009/M17049'.

Park, J., Kim, H., Tai, Y.-W., Brown, M. S. & Kweon, I. S. (2014), 'High-quality depth map upsampling and completion for rgb-d cameras', *IEEE Transactions on Image Processing* **23**(12), 5559–5572.

Prieto, M. S. & Allen, A. R. (2003), 'A similarity metric for edge images', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **25**(10), 1265–1273.

Riegler, G., Ferstl, D., Rüther, M. & Bischof, H. (2016), A deep primal-dual network for guided depth super-resolution, *in* 'Proceedings of the British Machine Vision Conference'.

Sang-Beom Lee, C. L. & Ho, Y.-S. (n.d.), Experimental results on improved temporal consistency enhancement, *in* 'ISO/IEC JTC1/SC29/WG11 MPEG2009/M16063'.

Schuon, S., Theobalt, C., Davis, J. & Thrun, S. (2009), Lidarboost: Depth superresolution for tof 3d shape scanning, *in* 'Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on', IEEE, pp. 343–350.

Silberman, N., Hoiem, D., Kohli, P. & Fergus, R. (2012), 'Indoor segmentation and support inference from rgbd images', *Computer Vision–ECCV 2012* pp. 746–760.

Szeliski, R., Zabih, R., Scharstein, D., Veksler, O., Kolmogorov, V., Agarwala, A., Tappen, M. & Rother, C. (2008), 'A comparative study of energy minimization methods for markov random fields with smoothness-based priors', *IEEE transactions on pattern analysis and machine intelligence* **30**(6), 1068–1080.

*tensorflow [Online]* (n.d.), `http://www.tensorflow.org`.

Tomasi, C. & Manduchi, R. (1998), Bilateral filtering for gray and color images, *in* 'Computer Vision, 1998. Sixth International Conference on', IEEE, pp. 839–846.

Tomioka, T., Mishiba, K., Oyamada, Y. & Kondo, K. (2016), Depth map estimation using census transform for light field cameras, *in* 'Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on', IEEE, pp. 1641–1645.

Veksler, O. (2003), Fast variable window for stereo correspondence using integral images, *in* 'Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on', Vol. 1, IEEE, pp. I–I.

Wang, Z., Liu, D., Yang, J., Han, W. & Huang, T. (2015), Deep networks for image super-resolution with sparse prior, *in* 'The IEEE International Conference on Computer Vision (ICCV)'.

Xie, J., Chou, C.-C., Feris, R. & Sun, M.-T. (2014), Single depth image super resolution and denoising via coupled dictionary learning with local constraints and shock filtering, *in* 'Multimedia and Expo (ICME), 2014 IEEE International Conference on', IEEE, pp. 1–6.

Xie, J., Feris, R. S. & Sun, M.-T. (2016), 'Edge-guided single depth image super resolution', *IEEE Transactions on Image Processing* **25**(1), 428–438.

Yaghoobi, M., Nam, S., Gribonval, R. & Davies, M. E. (2011), Analysis operator learning for overcomplete cosparse representations, *in* 'Signal Processing Conference, 2011 19th European', IEEE, pp. 1470–1474.

Yang, J., Ye, X., Li, K., Hou, C. & Wang, Y. (2014), 'Color-guided depth recovery from rgb-d data using an adaptive autoregressive model', *IEEE Transactions on Image Processing* **23**(8), 3443–3458.

Yang, Q. (2015), 'Stereo matching using tree filtering', *IEEE transactions on pattern analysis and machine intelligence* **37**(4), 834–846.

Yang, Q. & Ahuja, N. (2012), 'Stereo matching using epipolar distance transform', *IEEE Transactions on Image Processing* **21**(10), 4410–4419.

Yang, Q., Ahuja, N., Yang, R., Tan, K.-H., Davis, J., Culbertson, B., Apostolopoulos, J. & Wang, G. (2013), 'Fusion of median and bilateral filtering for range image upsampling', *IEEE Transactions on Image Processing* **22**(12), 4841–4852.

Yang, Q., Ji, P., Li, D., Yao, S. & Zhang, M. (2014), 'Fast stereo matching using adaptive guided filtering', *Image and Vision Computing* **32**(3), 202–211.

Yang, Q., Wang, L., Yang, R., Stewénius, H. & Nistér, D. (2009), 'Stereo matching with color-weighted correlation, hierarchical belief propagation, and occlusion handling', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **31**(3), 492–504.

Yang, Q., Yang, R., Davis, J. & Nistér, D. (2007), Spatial-depth super resolution for range images, *in* '2007 IEEE Conference on Computer Vision and Pattern Recognition', IEEE, pp. 1–8.

Yoon, K.-J. & Kweon, I. S. (2006), 'Adaptive support-weight approach for correspondence search', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **28**(4), 650–656.

Zhang, F., Dai, L., Xiang, S. & Zhang, X. (2015), Segment graph based image filtering: Fast structure-preserving smoothing, *in* 'Proceedings of the IEEE International Conference on Computer Vision', pp. 361–369.

Zhang, K., Lu, J. & Lafruit, G. (2009), 'Cross-based local stereo matching using orthogonal integral images', *IEEE Transactions on Circuits and Systems for Video Technology* **19**(7), 1073–1079.

Zhang, K., Lu, J., Lafruit, G., Lauwereins, R. & Van Gool, L. (2009), Robust stereo matching with fast normalized cross-correlation over shape-adaptive regions, *in* 'Image Processing (ICIP), 2009 16th IEEE International Conference on', IEEE, pp. 2357–2360.

Zhang, Q., Cui, C. H., Ngan, K. N. & Liu, Y. (2012), Depth estimation and view synthesis for narrow-baseline video, *in* 'Circuits and Systems (ISCAS), 2012 IEEE International Symposium on', IEEE, pp. 1883–1886.

Zhu, J., Wang, L., Gao, J. & Yang, R. (2010), 'Spatial-temporal fusion for high accuracy depth maps using dynamic mrfs', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **32**(5), 899–909.

Zuo, Y., Wu, Q., Zhang, J. & An, P. (2018), 'Explicit edge inconsistency evaluation model for color-guided depth map enhancement', *IEEE Transactions on Circuits and Systems for Video Technology* **28**(2), 439–453.