

Dissertation Title Typed in Upper and Lower Case Letters

by

Your Name [no degree credentials]

Submitted in Partial Fulfillment of the
Requirements for the Degree
Doctor of Philosophy

Supervised by Professor [Your advisor's first and last name]

[Department if applicable]

[School Name 1]

[School Name 2]

University of Rochester

Rochester, New York

2014 [or year of degree conferral]

Dedication goes here.

Biographical Sketch

The author was born in CITY, STATE. HE/SHE attended XXXX University, and graduated with a Bachelor of ARTS/SCIENCE degree in MAJOR FIELD. Describe Master's degree if from an institution other than the University of Rochester. Describe professional or career path prior to doctoral study if applicable. HE/SHE began doctoral studies in FIELD at the University of Rochester in 200X. HE/SHE was awarded [if applicable: a NAME(s) Fellowship in 200X and 200X] and [if applicable] received the Master of ARTS/SCIENCE degree from the University of Rochester. HE/SHE pursued HIS/HER research in SUBJECT under the direction of ADVISOR NAME.

The following publications were a result of work conducted during doctoral study: [list full bibliographic reference information in the format used elsewhere in the dissertation]

Abstract

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Prae-

sent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Contributors and Funding Sources

This work was supported by a dissertation committee consisting of Professor XXXX (advisor — also note if co-advisor) and XXXX of the Department of (Home Department) and Professor(s) XXXX of the Department of (Outside Department). The data analyzed for Chapter 3 was provided by Professor XXXX. The analyses depicted in Chapter 4 were conducted in part by Rebecca Jones of the Department of Biostatistics and were published in (year) in an article listed in the Biographical Sketch. All other work conducted for the dissertation was completed by the student independently. (If applicable) Graduate study was supported by a Provost Fellowship from the University of Rochester and a dissertation research fellowship from XXX Foundation.

Contents

1	Introduction	1
1.1	Background on depth map	2
1.2	Background on haze and dehazing	4
1.3	Contribution and future work	5
1.A	Appendix to Chapter ??	6
2	A Local-Linear-Fitting-Based Matting Approach for Accu- rate Depth Upsampling	10
2.1	Introduction	10
2.2	Related work	11
2.3	Proposed local-linear-fitting-based depth upsampling algorithm	13
2.3.1	Local-linear-fitting-based Problem Formulation	13
2.3.2	Optimization Solution	17
2.3.3	memory saving implementation	19
2.4	experimental results	24
2.4.1	Qualitative and Quantitative Results	25

2.4.2	Discussion	31
2.5	Conclusion	32
2.A	Appendix to Chapter 2.3.2	33
2.B	Appendix to Chapter 2.3.3	34
3	Hazerd: an outdoor scene dataset and benchmark for Single image dehazing	35
3.1	Introduction	35
3.2	physical model for haze	38
3.3	Haze simulation	41
3.4	dataset	43
3.5	benchmark: algorithms and datasets	46
3.6	discussion	55
4	Conclusion	58
	Bibliography	60

List of Tables

2.1	Quantitative comparison of different algorithms tested on Middlebury dataset [1]. The results is evaluated as MAE (the smaller the better) for four different sample rates, as listed in the second row. All values are computed based on the disparity maps. The best result in each situation is in bold font. . . .	30
2.2	Similarity and difference of the proposed algorithm with Levin's method [?]	32
3.1	Particle sizes for different weather	40
3.2	The visual range in HazeRD, and the corresponding weather condition and the scattering coefficient β	43

3.3	Performance of different dehazing methods on HazeRD, Middlebury, and NYU datasets. Each numerical entry is represented as the average over the images in the dataset \pm the standard deviation. The best performing algorithm for each dataset is indicated in bold font, and * in the Middlebury and NYU datasets indicates cases where the difference with respect to HazeRD was not statistically significant (5% level).	54
-----	---	----

List of Figures

1.1	Left:cubic structure, right: depth map(nearer is darker). . . .	3
1.2	Top: Time-of-Flight camera: low resolution; bottom: Structured light: missing points by occlusion.	7
1.3	Example depth maps.Top: resolution difference; bottom: Missing points.	8
1.4	The same scene in different weather: left: in hazy weather; right: in clear weather.	9
2.1	The intuitive interpretation for the proposed algorithm. Two patch, respectively cropped from the handle area and the box ara, display different patterns in depth maps. The intuition of the proposed method aims to construct a linear interpolation based on only pixels with the same object which is supposed to be similar in both depth and color.	15

- 2.2 The illustration of problem formulation in 1D. The magenta and cyan points show different color pixels in the patch of color image, and the circles around data points indicate the available low resolution depth values. The filled and un-filled circles mean the desired upsampled depth map and the input depth map, respectively, and the line is the fitting result on the example area. The weights are illustrated by the size of filled circles. 16
- 2.3 Qualitative results of the hole filling ability of our algorithm, tested on Middlebury stereo dataset 2014 [2]. The first column shows the input high resolution color images and the second column shows the corresponding depth maps. The results are shown in the last column. The images are processed at a low resolution of approximate 60k to 80k pixels. 26
- 2.4 Visual comparison of different algorithm tested on Middlebury dataset [1] at $4\times$ upsampling rate. Column from left to right: RGB images, ground truth, and the results of: bilinear, bicubic, IBL [3], TGV [4] and proposed; row from top to bottom: Art, Books, and Moebius. 27

2.5	Visual comparison of different algorithm tested on Middlebury dataset [1] at $4\times$ upsampling rate. Column from left to right: RGB images, ground truth, and the results of: bilinear, bicubic, IBL [3], TGV [4] and proposed; row from top to bottom: Art, Books, and Moebius.	28
2.6	Visual comparison of different algorithm tested on Middlebury dataset [1] at $4\times$ upsampling rate. Column from left to right: RGB images, ground truth, and the results of: bilinear, bicubic, IBL [3], TGV [4] and proposed; row from top to bottom: Art, Books, and Moebius.	29
3.1	The physical model for haze. Light at the sensor is composed of airlight scattered by particles suspended in the air and light reflected from imaged objects, which is attenuated when propagating through the air.	39
3.2	Flow for haze simulation based on (3.3).	44
3.3	Benchmarking workflow for evaluating dehazing algorithms using the HazeRD(proposed) and D-Hazy [5] dataset.	46
3.4	HazeRD Samples. From left to right: a hazy image, with the visual range of 50m, 100m, 200m, and 500m respectively. . . .	46
3.5	Histograms of 3 dataset. Left: HazeRD, Middlebury, and NYU, right: the Middlebury dataset and NYU.	47

-
- 3.6 Example of algorithms' performances. (a): a haze free image from HazeRD, and the results of dehazing of a corresponding hazy image obtained with: (b):He [6], (c): Meng [7], (d): Zhu [8], (e): Berman [9], (f): Cai [10], and (g): Ren [11]. 50
- 3.7 Example of algorithms' performances. (a): the ground truth transmission for the haze free image in 3.6, and the transmission estimates corresponding to the estimated images obtained with: (b):He [6], (c): Meng [7], (d): Zhu [8], (e): Berman [9], (f): Cai [10], and (g): Ren [11]. 51
- 3.8 Example of algorithms' performances. (a): a haze free image from Middlebury dataset, and the results of dehazing of a corresponding hazy image obtained with: (b):He [6], (c): Meng [7], (d): Zhu [8], (e): Berman [9], (f): Cai [10], and (g): Ren [11]. 52
- 3.9 Example of algorithms' performances. (a): a haze free image from NYU dataset, and the results of dehazing of a corresponding hazy image obtained with: (b):He [6], (c): Meng [7], (d): Zhu [8], (e): Berman [9], (f): Cai [10], and (g): Ren [11]. 53

- 3.10 Scatter diagram of one scene of the dark channel prior error (y axis) and the transmission error (x axis) in log scale. Each point represents the density of pixels with similar prior-transmission ratios. See the bright line. The dark prior channel is above 0.2 in most area, which is contradictory to the assumption. The transmission error is almost linearly related to the dark channel error. From left to right, fog: dense, thick, moderate, light. 57

Chapter 1

Introduction

Traditional images contain informations of colors of some scenes, and when stored as typical image formats in computers, these images usually are converted to some specific color spaces for the convenience of displayers, for example, jpeg format use s-RGB space. This interpretation of images provides the fundamental of most image processing tasks and their solutions, from simple tasks such as edge detection, to difficult tasks such as scene understanding.

With the development of technologies, for present days it is possible to get more informations other than merely color. In this thesis we will mainly focus on depth informations, which will be referred to as *depth map*. *Depth maps* are images containing information relating to the distance of the surfaces of scene objects from a viewpoint. The depth information directly provides the clue for the spacial relation of the objects in some scene, which is vital in many image processing problems, such as semantic segmentation and object recognition. At the same time, the usage of depth map introduces new tasks:

the acquisition of high quality depth map, the registration of corresponding depth map and color images, etc.

In this thesis we will study the both sides of the utilization of depth maps: in chapter 2, we propose a local-linear fitting based depth upsampling algorithm to obtain high resolution and accurate depth map, as well as overcoming the intrinsic missing points problem; and in chapter 3, we exploit the depth map to construct a synthesized outdoor scene dataset obeying the atmospheric model, for the benchmarking of image dehazing algorithms.

This chapter is organized as follow: section 1.1 provides the background on the acquisition and defects of depth maps; section 1.2 discusses the motivation for image dehazing problem; section 1.3 discusses the contribution of this thesis and the future work.

Background on depth map

Depth map with corresponding RGB images can be called *RGB-D images*. Figure. 1.1 shows an example of one pair of RGB-D images. RGB-D images play an important role in image processing and machine vision: in stereo researches depth maps serve as the centre information for inferring 3D reconstructions and registration; in other core tasks in image processing and machine vision, they are also widely used, for example segmentation, tracking and image dehazing. Typical techniques to obtain depth maps includes Time-Of-Flight(ToF) camera, structured light based systems, and Lidar scanning

systems generating points cloud from which the depth can be extracted.

The key challenge in using RGB-D images is that the depth images are limited in resolution compared to corresponding RGB images, and that the generated depth maps inevitably have missing points. The two prominent methods for capturing depth data are time-of-Flight (ToF) and structured light based systems. While ToF based systems provide highly accurate depth information, they are relatively tedious to use and even after sophisticated alignment with images [12], typically offer a lower resolution than typical high resolution color cameras. For structured light based RGB-D images a significant fraction of the pixels (up to 10%) are not assigned depth values due to the challenges of these systems. Thus for both ToF and structured light based RGB-D image capture systems, some forms of depth upsampling (including hole filling) are required to generate a complete RGB-D image. Figure. 1.2 and 1.3 depict the principle for the two capturing methods and the drawbacks. In this thesis we want to propose a depth upsampling algorithm to overcome both difficulties.

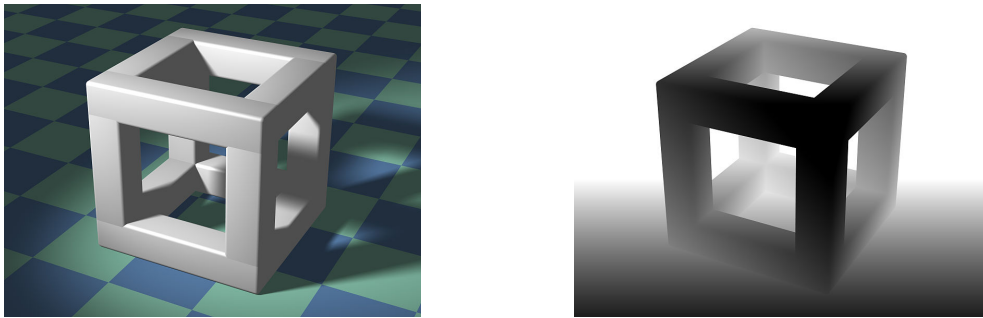


Figure 1.1: Left:cubic structure, right: depth map(nearer is darker).

Background on haze and dehazing

Haze and fog are common atmospheric phenomena, caused by the high density and various size of water droplets or particles distributed in the air. Both human vision and images taken under such condition are limited in hazy weather, on the clarity of the contour and the blurred color of objects, especially when the objects are far away or the haze is dense. Figure 1.4 is an example for the same scene in hazy weather and clear weather: the color in hazy part is generally more "gray" than the normal part, and the edges of different objects are more difficult to tell. This degradation may results in significant problems in many areas, for example when driving in hazy days, the velocity must be slower than clear days to keep safe. Due to the similarity in physical model, under water vision also has the analogous effects. This problem introduces a new task for image processing, namely *image dehazing*, or eliminating the haze in images. Another reason driven this study on dehazing algorithms is that air and haze caused by air pollution has attract more and more attentions all over the world, especially in China. We hope the study can do some help in alleviating this problem.

There are many dehazing algorithms being proposed by now, either based on certain assumptions or other machine learning frameworks. However, there are no general criteria for evaluating the performances of different algorithms. And for existing datasets, most of them are indoor scenes which is not the typical application situation for dehazing algorithms. To build a

synthesized hazy scene involves both color images and depth informations, which is a suitable application for RGB-D images. Thus we want to construct a dataset for image dehazing problem, providing outdoor scenes of different weather conditions and ground truth.

Contribution and future work

This thesis has two main contributions. First, an accurate depth map upsampling algorithm is proposed. This method is based on local linear fitting and utilizes the color information to improve the edge quality. We also propose a memory saving implementation to solve the optimization problem. Second, we use RGB-D images to construct an outdoor scene dataset, *HazeRD*, and benchmarked some state-of-the-art dehazing algorithms. Compared to existing dataset, *HazeRD* provides high-resolution outdoor scenes of different haze levels with ground truth, instead of indoor scenes, which is the typical conditions for the application of dehazing algorithm.

The work in this thesis also has some limitations. The proposed depth upsampling algorithm is extremely time consuming compared to naive methods such as bilinear or bicubic interpolation. In future work, one direction is to use parallel techniques to accelerate the computing. Besides, the benchmarking results of state of art dehazing algorithms indicates that the main bottleneck lies in the consistency of the estimated transmission of the same objects. In future work, new dehazing algorithms combining image segmen-

tation algorithms should be considered.

Appendix to Chapter ??

Sometimes you might want a chapter-specific appendix numbered separately from the section numbers in the main text. This is especially useful when you are dealing with numbered equations or theorems, and want a way to denote that a result lives in the appendix. See the source code for this chapter for how this is done.

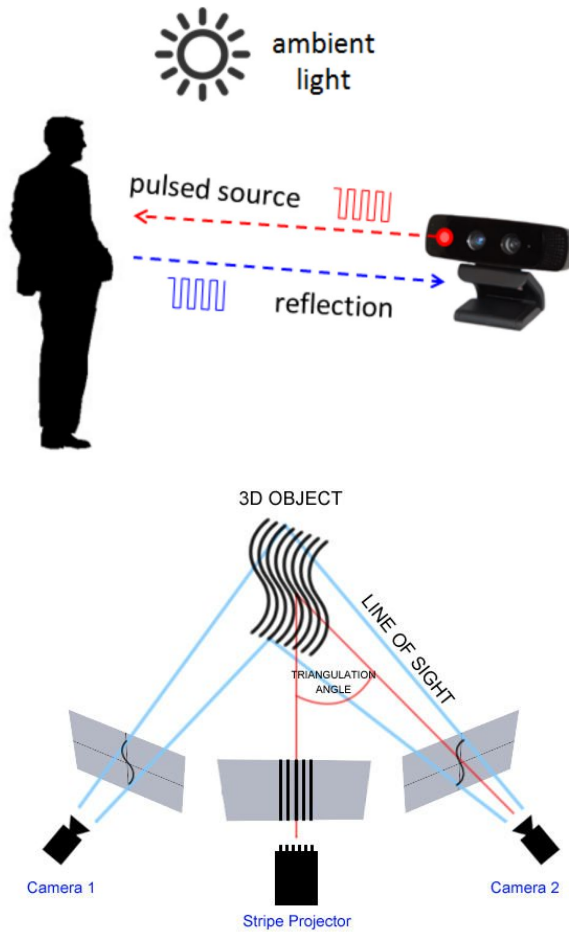


Figure 1.2: Top: Time-of-Flight camera: low resolution; bottom: Structured light: missing points by occlusion.



Figure 1.3: Example depth maps. Top: resolution difference; bottom: Missing points.



Figure 1.4: The same scene in different weather: left: in hazy weather; right: in clear weather.

Chapter 2

A Local-Linear-Fitting-Based Matting Approach for Accurate Depth Upsampling

Introduction

Depth upsampling aims at obtaining high resolution depth maps based on low resolution version. Naive method such as bilinear or bicubic interpolation uses only depth informations, which result in the problem of blurred, gradual-changing edges, making the interpolation terrible for complex scenes. To overcome this problem, color image can be exploited to constraint the interpolation consistent within contours. For smooth area, simple bilinear methods usually can provide a satisfying results and advanced algorithms can behave similar to keep the procedure compact. We obey this observation and provide a local linear fitting based matting approach for accurate depth upsampling in this chapter.

This chapter will discuss a local-linear-fitting based matting approach for accurate depth upsampling, and the memory saving optimization method

using conjugate gradient solver. This chapter is organized as follows: Section 2.2 briefly discuss related literature to depth upsampling. Section 2.3 describes the scheme of our algorithm, and the memory saving solver. We present both the quantitative and the qualitative results in Section 2.4, and conclude the paper in Section 2.5.

Related work

Traditionally depth upsampling is accomplished by bilinear or bicubic interpolation. These methods have difficulty in preserving the sharp edges in depth maps. Several methods have been developed to overcome these problems, aiming at improving the accuracy of depth upsampling problem. One class of techniques relies on proposing a prior and optimizing an objective function that combines prior and data fidelity terms [13, 14, 15, 16, 17, 4, 3]. Diebel and Thrun [13] proposed an upsampling algorithm based on Markov random field (MRF), which is defined through depth measure potential, depth smooth prior and weighting factors. This MRF framework is further improved by other researchers, such as [18] and [19]. Yang et al. [14] made use of a bilateral filter in an iterative refinement framework. The refinement is constructed on a cost volume defined on the current depth map and the RGB image. This algorithm can also work on two view depth map refinement with a different cost volume definition. In [15], the guided filter was designed for edge preserving filter, which can be viewed as an extension of

the bilateral filter. Kopf et al. [16] proposed joint a bilateral filter which is also similar in principle. Both filters can be used to upsample the depth map with a high resolution RGB image. Park et al. [17] gave an algorithm based on a non local mean filter. The low resolution depth map is pre-processed to detect outliers. These points are removed and to obtain the high resolution depth map an objective function consisting of a smooth term, non-local structure term and data term is optimized. This algorithm is also suitable for filling large holes in the depth data. Ferstl et al. [4] gave an algorithm based on total generalization variance (TGV). A TGV regularization weighted according to intensity image texture is used in the objective function and the optimization is solved as a primal-dual problem. Yang et al. [3] built a color-guided adaptive regression model for depth map upsampling. Different edge preserving terms including non-local mean and bilateral filters are tested and an analysis is given on the parameter selection and the system stability.

Another category of depth map upsampling utilizes segmentation techniques to extract depth information. Krishnamurthy and Ramakrishnan [20] and Uruma et al. [21] start from an upsampled depth map using standard interpolation methods and refine the result by image segmentation techniques. The segmentation process serves a similar function in preserving edges as the afore-mentioned filters.

A common theme of prior algorithms, also adopted in our work, is to "fix" edges of upsampled depth map for better consistency with the color image. Our work is inspired by Levin et al.'s optimization formulation of

matting [22], in which the alpha value for the matting mask is modeled as a linear combination of neighboring color values. Analogous to the matting problem, we formulate depth upsampling as an optimization problem. Specifically, the upsampled image is estimated by minimizing an objective function comprising two additive terms. The first term ensures that the estimated depth map is locally smooth consistent with the color image and the second term ensures consistency of the estimated upsampled data with the low resolution observed data at the corresponding locations. Depth map upsampling is then achieved by solving a large sparse linear system following a similar approach as was done for matting in [22]. A key difference between the matting problem and our approach is that we model the depth as a *linear function of the local spatial coordinates* and not as a linear function of the image intensity values.

Proposed local-linear-fitting-based depth upsampling algorithm

Local-linear-fitting-based Problem Formulation

Our proposed method is motivated by the fact that regions of the image that correspond to a smooth 3D surface, can be locally approximated by a plane (for example, via a Taylor series expansion). Thus, over each small patch in the image in regions corresponding to smooth surfaces, a local linear fit

(in spatial coordinates) provides a good approximation to the depth. To account for edges, where the assumption breaks down, adaptive nonnegative weights are introduced for the linear fitting. The weighting seeks to effectively concentrate the linear fit at each point on the neighboring pixel locations that are hypothesized, based on their color similarity to the pixel of interest, to be on the same side of the edge. The weights can be obtained from one of several edge preserving techniques, for example, non local mean or bilateral filter. The upsampled depth map is obtained by minimizing an overall objective function that combines a term corresponding to the weighted deviation from the local linear fitting with a data fidelity term that penalizes deviations from observations at the locations where the low resolution depth map is available. An intuitive interpretation is depicted in Fig. ??.

To formally describe our algorithm we use the simplified 1D representation in Fig. 2.2 that illustrates the contribution of one pixel to the objective function. The axis G represents the relative pixel positions of points in local pixel neighborhood of the target pixel which is located at $G = 0$. The low resolution depth map, denoted by D_L , is available at a subset of the pixel locations in the neighborhood as indicated in the figure and color values, denoted by I , form the high resolution RGB image. The goal is to estimate a high resolution depth map D_H . Our objective function is formulated as

$$Q = \sum_{j=1}^N \sum_{i \in \mathcal{N}(j)} ||w_{i,j}(d_{i,j} - D_{H,i})||^2 + \sum_{j=1}^M \lambda(D_{L,j} - D_{H,j})^2, \quad (2.1)$$

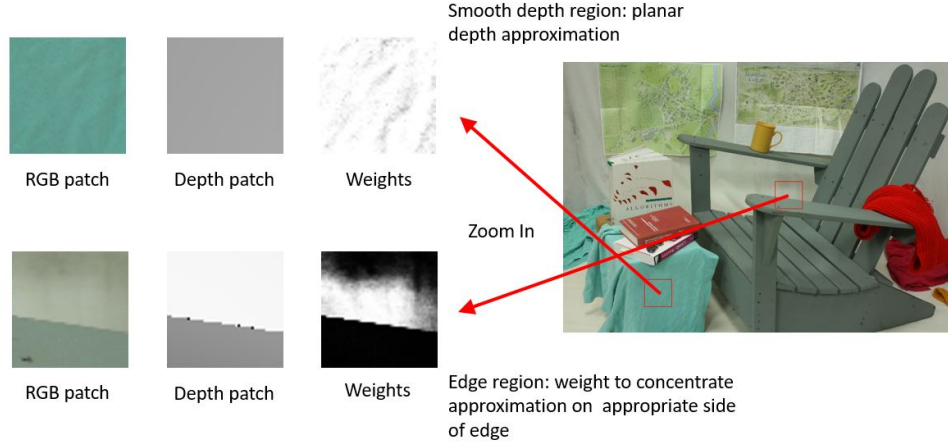


Figure 2.1: The intuitive interpretation for the proposed algorithm. Two patch, respectively cropped from the handle area and the box area, display different patterns in depth maps. The intuition of the proposed method aims to construct a linear interpolation based on only pixels with the same object which is supposed to be similar in both depth and color.

where j indexes the pixel locations in the upsampled image, N is the number of pixels in the upsampled image, M is the number of pixels in the low resolution depth map, $d_{i,j}$ is the value of linear fitting of pixel i in the neighbor of pixel j , $D_{H,i}$ is the estimated depth at pixel i in the neighbor of pixel j , and $D_{L,j}$ is the depth value at the pixel j of the low resolution depth map, $w_{i,j}$ is the similarity metric of pixel i and j , and λ is the free parameter to control the relation of fidelity and smoothness. The local linear fit is defined as

$$d_{i,j} = A_j G_{i,j} + B_j, \quad (2.2)$$

where A_j and B_j are the parameters for linear fitting at pixel $D_{H,j}$, A_j is a 1-by-2 vector and B_j is a scalar, and $G_{i,j}$ is a 2-by-1 vector denoting the

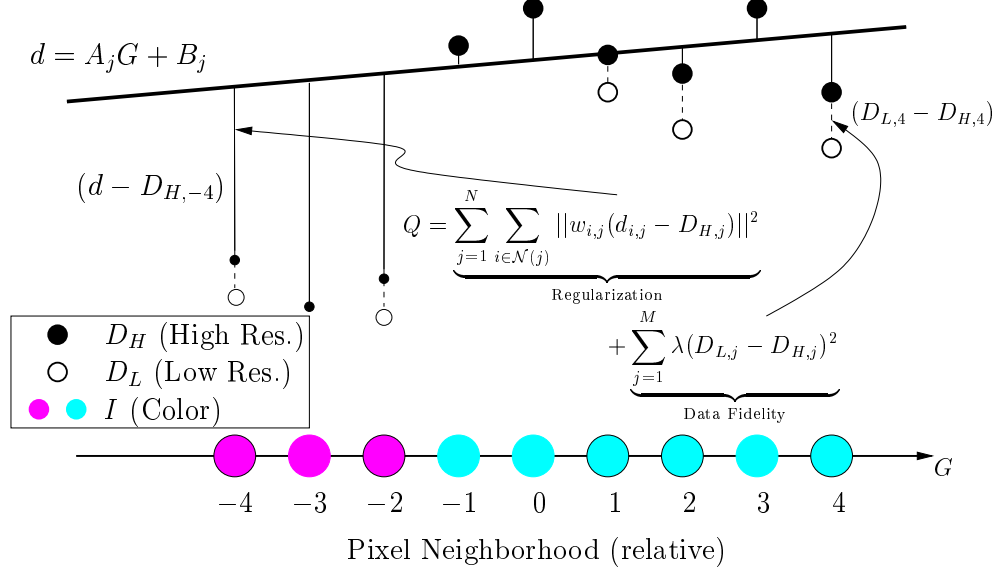


Figure 2.2: The illustration of problem formulation in 1D. The magenta and cyan points show different color pixels in the patch of color image, and the circles around data points indicate the available low resolution depth values. The filled and un-filled circles mean the desired upsampled depth map and the input depth map, respectively, and the line is the fitting result on the example area. The weights are illustrated by the size of filled circles.

relative coordinate of pixel i in the neighbor window of pixel j . We define $G_{i,j} \stackrel{\text{def}}{=} \{(x, y) | -w_s < x < w_s, -w_s < y < w_s\}$, where w_s is the size of the window. The first term in (??) is the regularization term and the second term is the data fidelity. The formulation is readily extended to the hole filling problem by adding, to the fidelity penalty term, a product with the indicator function of non-missing points and pixel values.

The weights $w_{i,j}$ are defined as

$$w_{i,j} = \exp - \frac{\|I_i - I_j\|^2}{2\sigma^2}, \quad (2.3)$$

where I_i and I_j are pixel values of the RGB image I at corresponding position, and σ controls the relative emphasis of pixel similarity in the allocation of weights. Alternative, formulations of the weights such as those used in non local mean or bilateral filter can also be used in the proposed framework. Unlike the typical bilateral filter, we do not use the distance decay term in (2.3) because the window we use is quite small comparing to the high resolution images.

Our problem formulation and the algorithmic approach we use for the solution (described in the next section) are inspired by Levin’s formulation of matting as an optimization problem [22], where the alpha channel is formulated as a weighted linear combination of neighboring color values. A key difference in our formulation is that the our weighted local linear fit is formulated in terms of the local relative *spatial* position for the neighborhood, whereas in [22] the weighted linear fit is performed on the *color* values for the neighborhood pixels.

Optimization Solution

Rewriting (2.1) in the matrix form, we obtain

$$Q = \sum_{j=1}^N (W_j(D_{H,N_j} - GP_j^T))^2 + \lambda F_j, \quad (2.4)$$

where $G = [G_j, 1]$ and $P_j = [A_j, B_j]^T$. W_j is a diagonal matrix with $w_{i,j}$ being its diagonal entries. D_{H,N_j} is the depth value in the patch¹. The matrix P_j can be eliminated by replacing it in (2.4) by its optimal value

$$\begin{aligned} P_j &= \operatorname{argmin}_{P_j} ((W_j(D_{H,N_j} - GP_j^T))^2) \\ &= (G^T W_{0,j}^T G)^{-1} G^T W_{0,j}^T D_{H,N_j}, \end{aligned} \quad (2.5)$$

where $W_{0,j}$ is the diagonal matrix,

$$W_{0,j} = W_j^T W_j, \quad (2.6)$$

Replacing P_j in (2.1) by (2.5), we obtain,

$$Q = \sum_{j=1}^N D_{H,N_j}^T (\overline{G_j}^T W_{0,j} \overline{G_j}) D_{H,N_j} + \sum_{j=1}^M \lambda (D_{L,j} - D_{H,j})^2, \quad (2.7)$$

$$\overline{G_j} = E - G(G^T W_{0,j}^T G)^{-1} G^T W_{0,j}, \quad (2.8)$$

where E denoting the identity matrix.

The minimizer for the quadratic objective function Q is readily obtained, specifically, as the solution to the linear equation,

$$LD + \lambda A(D - d) = 0, \quad (2.9)$$

¹We pad the image to represent G consistently at all positions.

where L is the Laplacian matrix [23],

$$L = \sum_{j=1}^N \overline{G_j}^T W_{0,j} \overline{G_j} \quad (2.10)$$

and A is a diagonal matrix indicating the correspondence of pixels in low resolution map to the upsampled map. The derivation detail is listed in appendix 2.A.

memory saving implementation

In last section the optimization of the objective is formulated as an large sparse linear system, which can be efficiently solved with methods such as conjugate gradient solver. In practise, one important constraint for this method is that the Laplacian matrix and the successive processing, even though sparse, are very large and require a lot of memory. To reduce the memory requirement, we propose a computational efficiency improvement for the optimization, still exploiting the framework of conjugate gradient method [24, 25].

Algorithm 1 describes the conjugate gradient algorithm for solving sparse linear systems. This algorithm is designed for solving symmetric and positive-definite linear systems, which is suitable for the optimization in our problem where the Laplacian matrix automatically satisfies the required condition. This method exploits the conjugate vectors with respect to \mathbf{A} and iteratively approximates the closest solution.

input : Initial guess \mathbf{I}_0 , convergence threshold τ

output : $\tilde{\mathbf{I}}$: estimate for \mathbf{I}

Procedure Initialize: $\tilde{\mathbf{I}} \leftarrow \mathbf{I}_0$, $\mathbf{r}_0 \leftarrow \mathbf{b} - \mathbf{A}\tilde{\mathbf{I}}$, $\mathbf{p}_0 \leftarrow \mathbf{r}_0$, $j \leftarrow 0$;

while $\mathbf{r}_j^T \mathbf{r}_j > \tau \|\mathbf{I}\|$ **do**

$$\alpha_j \leftarrow \frac{\mathbf{r}_j^T \mathbf{r}_j}{\mathbf{p}_j^T \mathbf{A} \mathbf{p}_j};$$

$$\tilde{\mathbf{I}} \leftarrow \tilde{\mathbf{I}} + \alpha_j \mathbf{p}_j;$$

$$\mathbf{r}_{j+1} \leftarrow \mathbf{r}_j - \alpha_j \mathbf{A} \mathbf{p}_j;$$

$$\beta_j \leftarrow \frac{\mathbf{r}_{j+1}^T \mathbf{r}_{j+1}}{\mathbf{r}_j^T \mathbf{r}_j};$$

$$\mathbf{p}_{j+1} \leftarrow \mathbf{r}_{j+1} + \beta_j \mathbf{p}_j;$$

end

Algorithm 1: Solve the sparse linear system $\mathbf{A}\mathbf{I} = \mathbf{b}$ using conjugate-gradient algorithm.

One bottleneck for the proposed depth upsampling algorithm is that, for high resolution images, the Laplacian matrix is beyond typical computers' memory, for example a mega-pixel image will require constructing a matrix of tera-entries. This problem can be effectively solved with some modifications on the naive conjugate gradient method, namely calculating $\mathbf{A}\mathbf{p}_j$ in algorithm 1 without explicitly constructing \mathbf{A} , in our case the Laplacian matrix.

From (2.10) we can obtain the entry $L_{i,j}$ at the i th row and the j column of the Laplacian matrix,

$$L_{i,j} = \sum_{k|i, j \in N_k} (\delta_{ij}w_{ki} - w_{ki}w_{kj}(G_i - k_0)C_k(G_j - k_0)), \quad (2.11)$$

where N_k is the neighbour of pixel k , whose range is specified by the window size, w_{ki} is the weight of pixel k to pixel i and j , δ_{ij} is the Kronecker delta, C_k is the inverse of $\overline{G_j}^T W_{0,j} \overline{G_j}$ of pixel k , G_i is the relative coordinate of pixel i to k , and k_0 is the global coordinate of the pixel k . Then we break the summation stepwise, first computing,

$$a_k = C_k \left(\sum_{j \in N_k} w_{kj} G_j P_j - k \overline{p_k} \right), \quad (2.12)$$

where P_j is the i th entry of vector P , and $\overline{p_k}$ is the average of P_j in N_k . then,

$$b_k = k_0 a_k, \quad (2.13)$$

at the last step, combine a_k and b_k to obtain $(Lp)_i$, the entry at the i th

column of vector Lp ,

$$(Lp)_i = \sum \delta_{ij} w_{ki} P_i - G_i \sum_{k \in N_i} a_k w_{ki} + \sum_{k \in N_i} b_k w_{ki}, \quad (2.14)$$

further, the inverse C_k can be expressed as,

$$C = \begin{pmatrix} \mathbf{y}^T \mathbf{w} \mathbf{y} \mathbf{1}^T \mathbf{w} \mathbf{1} - \mathbf{y}^T \mathbf{w} \mathbf{1} \mathbf{1}^T \mathbf{w} \mathbf{y} & \mathbf{y}^T \mathbf{w} \mathbf{1} \mathbf{1}^T \mathbf{w} \mathbf{x} - \mathbf{y}^T \mathbf{w} \mathbf{x} \mathbf{1}^T \mathbf{w} \mathbf{1} & \mathbf{y}^T \mathbf{w} \mathbf{x} \mathbf{1}^T \mathbf{w} \mathbf{y} - \mathbf{y}^T \mathbf{w} \mathbf{y} \mathbf{1}^T \mathbf{w} \mathbf{x} \\ \mathbf{x}^T \mathbf{w} \mathbf{1} \mathbf{1}^T \mathbf{w} \mathbf{y} - \mathbf{x}^T \mathbf{w} \mathbf{y} \mathbf{1}^T \mathbf{w} \mathbf{1} & \mathbf{x}^T \mathbf{w} \mathbf{x} \mathbf{1}^T \mathbf{w} \mathbf{1} - \mathbf{x}^T \mathbf{w} \mathbf{1} \mathbf{1}^T \mathbf{w} \mathbf{x} & \mathbf{x}^T \mathbf{w} \mathbf{y} \mathbf{1}^T \mathbf{w} \mathbf{x} - \mathbf{x}^T \mathbf{w} \mathbf{x} \mathbf{1}^T \mathbf{w} \mathbf{y} \\ \mathbf{x}^T \mathbf{w} \mathbf{y} \mathbf{y}^T \mathbf{w} \mathbf{1} - \mathbf{x}^T \mathbf{w} \mathbf{1} \mathbf{y}^T \mathbf{w} \mathbf{y} & \mathbf{x}^T \mathbf{w} \mathbf{1} \mathbf{y}^T \mathbf{w} \mathbf{x} - \mathbf{x}^T \mathbf{w} \mathbf{x} \mathbf{y}^T \mathbf{w} \mathbf{1} & \mathbf{x}^T \mathbf{w} \mathbf{x} \mathbf{y}^T \mathbf{w} \mathbf{y} - \mathbf{x}^T \mathbf{w} \mathbf{y} \mathbf{y}^T \mathbf{w} \mathbf{x} \end{pmatrix}, \quad (2.15)$$

or more compactly as the minor of the matrix:

$$C = \begin{pmatrix} M_{1,1} & M_{1,2} & M_{1,3} \\ M_{2,1} & M_{2,2} & M_{2,3} \\ M_{3,1} & M_{3,2} & M_{3,3} \end{pmatrix}, \quad (2.16)$$

here the normalization term omitted.

Substitute the update step of p_{j+1} in algorithm 1 with (2.14) we can get the version for optimization the proposed depth upsampling algorithm. The derivation detail is listed in appendix 2.B.

In colorization problem the summation in (2.12), (2.13), and (2.14) can be efficiently calculated by integral image techniques and dynamic programming. This step is possible as that the formulation in colorization utilizes a fixed summation table; but in our case, the $G_j w_{kj}$ is a summed table of localized filters w_k , which means that the values re-used in summed table is no longer applicable here. This restriction limits the acceleration of computing but still allows the memory saving. By this implementation the required memory for a two mega-pixel image is reduced from 60GB to 2GB memory. With parallel computation, this method can have a good balance in memory usage and time consumption.

experimental results

We test our algorithm on the Middlebury (stereo) dataset [1, 26, 27, 2], which provides high resolution RGB images of multiple views and corresponding disparity maps, which are used as the ground truth in our experiment. We use a window size of 7×7 ($\equiv N = 49$), and $\lambda = 10^5$. The RGB-D images are zero-padded for consistent use of (??), and the padded area is cropped out in the final results. The parameter σ^2 in (??) for computation of the weights $w_{i,j}$ is set to one third of the local variance in each window. In each

patch, the weight of the center pixel is set to 10^{-5} . We use the built-in Matlab conjugate gradient solver (*cgs*) for solving (??) (a tolerance of 10^{-10} and maximum number of iteration 10^4 were used).

Qualitative and Quantitative Results

The proposed algorithm is both suitable for hole filling for single disparity map and depth map upsampling, as indicated earlier. In this part, we first visually examine the performance of filling holes in depth map, as shown in Fig. 2.3. From the images in the last column, we can find that the holes, which correspond to the occluded area in the disparity map, are well filled. Unlike traditional interpolation methods, our algorithm is able to fix the holes in the depth images, so as to keep the consistency of depth map edges with those in the RGB images and avoid smoothing in such areas. For example, see the third row of Fig. 2.3. The missing points along the wall are well fitted to the two sides, and not blurred as a large patch.

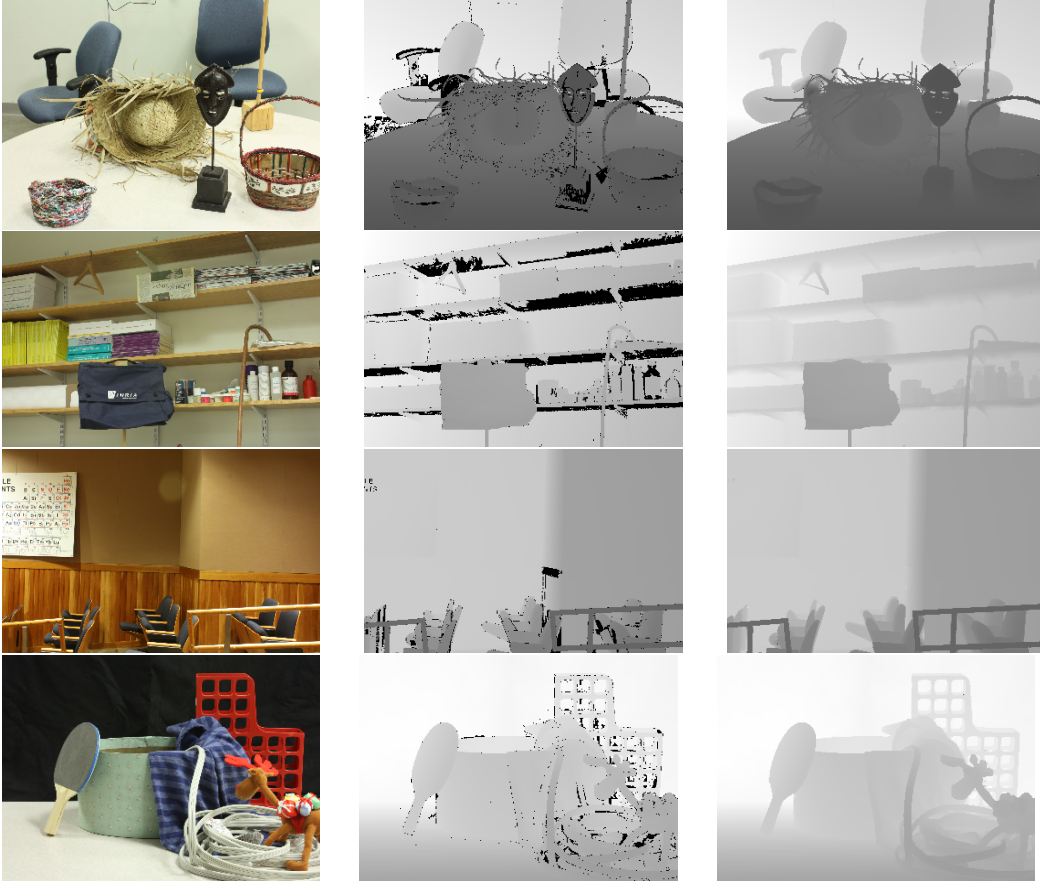


Figure 2.3: Qualitative results of the hole filling ability of our algorithm, tested on Middlebury stereo dataset 2014 [2]. The first column shows the input high resolution color images and the second column shows the corresponding depth maps. The results are shown in the last column. The images are processed at a low resolution of approximate 60k to 80k pixels.



Figure 2.4: Visual comparison of different algorithm tested on Middlebury dataset [1] at $4\times$ upsampling rate. Column from left to right: RGB images, ground truth, and the results of: bilinear, bicubic, IBL [3], TGV [4] and proposed; row from top to bottom: Art, Books, and Moebius.

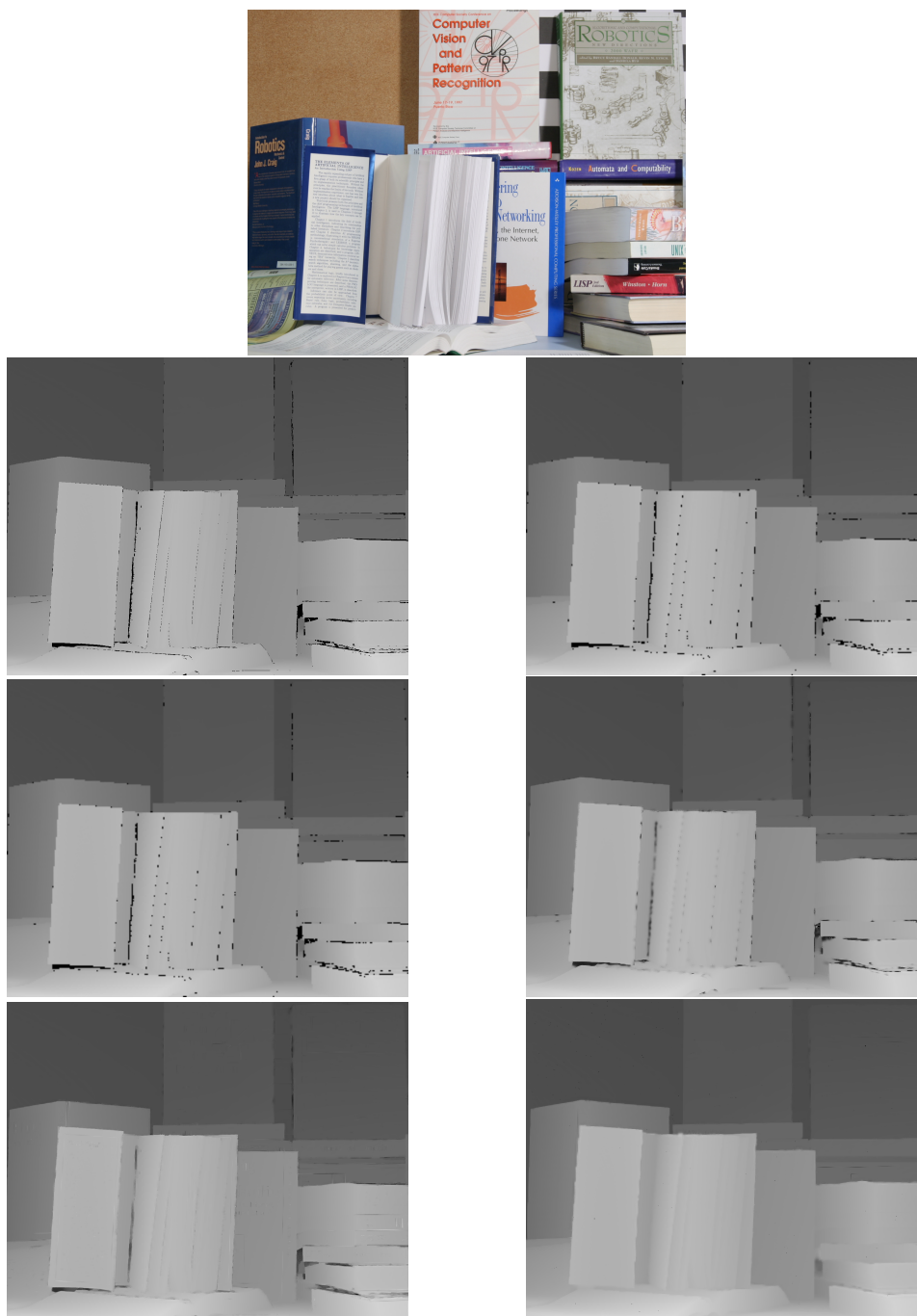


Figure 2.5: Visual comparison of different algorithm tested on Middlebury dataset [1] at $4\times$ upsampling rate. Column from left to right: RGB images, ground truth, and the results of: bilinear, bicubic, IBL [3], TGV [4] and proposed; row from top to bottom: Art, Books, and Moebius.

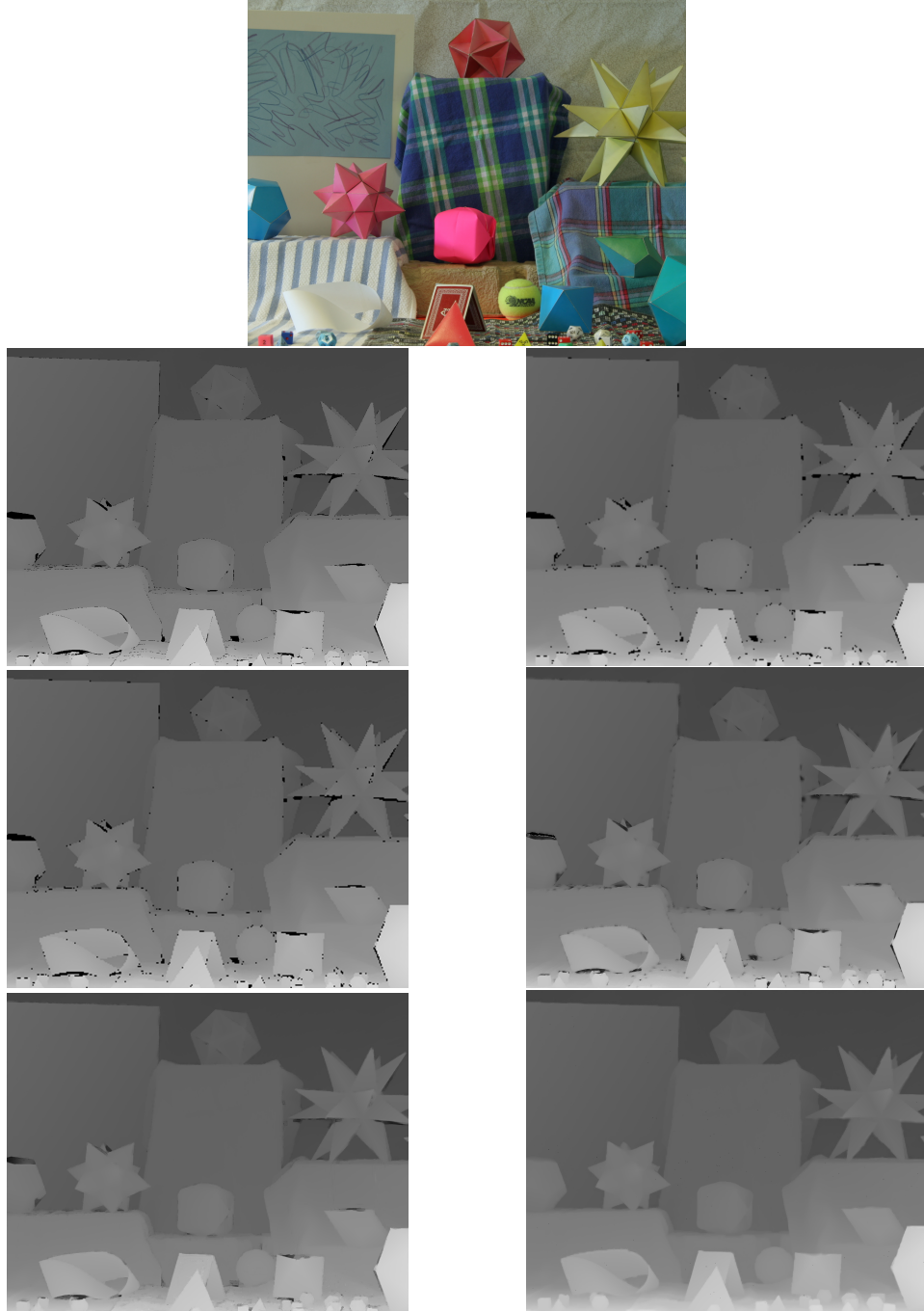


Figure 2.6: Visual comparison of different algorithm tested on Middlebury dataset [1] at $4\times$ upsampling rate. Column from left to right: RGB images, ground truth, and the results of: bilinear, bicubic, IBL [3], TGV [4] and proposed; row from top to bottom: Art, Books, and Moebius.

Sample Rate Methods	Images											
	Art				Books				Moebius			
	2×	4×	8×	16×	2×	4×	8×	16×	2×	4×	8×	16×
bicubic	0.8965	1.4298	2.4363	4.3456	0.7911	1.0842	1.7031	2.5419	0.6855	1.0287	1.5821	2.5527
bilinear	0.7642	1.2300	2.1495	3.9500	0.6620	0.8993	1.4183	2.1174	0.5685	0.8578	1.3347	2.1942
IBL [14]	0.5016	0.8934	1.7028	4.2324	0.2790	0.7361	1.4056	2.4561	0.3987	0.7071	1.1289	2.5885
TGV [4]	0.6457	0.8926	3.2633	7.6490	0.5980	0.7507	2.3091	6.324	0.4722	0.5627	2.0375	6.6210
Proposed	0.4423	0.8765	1.7616	3.6033	0.1986	0.3594	0.6655	1.1888	0.1864	0.3426	0.6478	1.2393

Table 2.1: Quantitative comparison of different algorithms tested on Middlebury dataset [1]. The results is evaluated as MAE (the smaller the better) for four different sample rates, as listed in the second row. All values are computed based on the disparity maps. The best result in each situation is in bold font.

Quantitative results comparing the proposed algorithm against prior work are obtained on the Middlebury stereo dataset 2005 [1]. We first downsample the input depth map to obtain the low resolution version, and then run different algorithms on these images to obtain the upsampled versions. We use mean absolute error (MAE) as the metric to evaluate the performance of different algorithms. Table 2.1 summarizes the results². Fig. 2.6 shows the corresponding visual results at the upsampling rate of 4. The results show that our algorithm is particularly suitable for depth map upsampling, and our algorithm provides a better result compared with the other algorithms.

Discussion

Table 2.1 illustrates that the depth map upsampling obtained with the proposed algorithm is accurate and achieves the state of the art results on the common benchmarking dataset, providing a better performance compared with other algorithms. The proposed algorithm, however, still suffers from two limitations. First, there are a few outlier points where the method yields a large error. Second, the edges are not sharply defined, especially under high upsampling rate, which is typical in most depth upsampling algorithms. The computational requirements are an additional challenge: to process a 1088×1296 pixel image, our algorithm takes about 40min. While the time requirement is analogous for several other upsampling algorithms, a speed-up

²Code for IBL implementation is provided by Chunhua Shen: <https://bitbucket.org/chhshen/depth-enhancement>.

is desirable for many applications.

We also want to mention more about the relation of the proposed algorithm with Levin’s [22] matting method and He’s [24] fast computing algorithm. Levin’s method is to formulate a linear combination of neighbour pixels value for estimating matting map, or alpha channel; He’s method is based on Levin’s formulation and proposed the acceleration also using conjugate gradient. In our problem, we utilize the linear relation of spacial instead direct pixel values; and, we formulate a local filtered conjugate gradient formulation using conjugate gradient, instead of an adjacent matrix. The similarity and difference of the proposed algorithm with Levin’s method is listed in 2.2.

	Levin’s matting method	Proposed method
similarity	<ul style="list-style-type: none"> • linear model • build Laplacian matrix to solve the optimization 	
difference	linear in color intensity	<ul style="list-style-type: none"> • linear in relative spatial location • weight based on color information

Table 2.2: Similarity and difference of the proposed algorithm with Levin’s method [?]

Conclusion

The algorithm proposed in this paper provides an effective method for depth map upsampling and hole filling. Quantitative results on test data indicates that the method offers an improvement over current state of the art methods and visual assessment shows that the depth map estimated by the proposed

technique is consistent with the color images.

Appendix to Chapter 2.3.2

For (2.5),

$$\begin{aligned} P_j &= \operatorname{argmin}_{P_j} ((W_j(D_{H,N_j} - GP_j^T))^2) \\ &= (G^T W_{0,j}^T G)^{-1} G^T W_{0,j}^T D_{H,N_j}, \end{aligned} \tag{2.17}$$

so,

$$\begin{aligned} d((W_j(D_{H,N_j} - GP_j^T))^2)/dP_j &= 0 \\ &= W_j(D_{H,N_j} - GP_j^T)G^T, \end{aligned} \tag{2.18}$$

solve the linear equation gives,

$$P_j = (G^T W_{0,j}^T G)^{-1} G^T W_{0,j}^T D_{H,N_j}. \tag{2.19}$$

For (2.20),

$$\begin{aligned}
 Q &= \sum_{j=1}^N (W_j(D_{H,N_j} - G(G^T W_{0,j}^T G)^{-1} G^T W_{0,j}^T D_{H,N_j}^T))^2 + \lambda F_j \\
 &= \sum_{j=1}^N ((W_j D_{H,N_j})(E - G(G^T W_{0,j}^T G)^{-1} G^T W_{0,j}^T))^2 + \lambda F_j \\
 &= \sum_{j=1}^N D_{H,N_j}^T (\overline{G_j}^T W_{0,j} \overline{G_j}) D_{H,N_j} + \lambda F_j \\
 &= \sum_{j=1}^N D_{H,N_j}^T (\overline{G_j}^T W_{0,j} \overline{G_j}) D_{H,N_j} + \sum_{j=1}^M \lambda (D_{L,j} - D_{H,j})^2,
 \end{aligned} \tag{2.20}$$

Appendix to Chapter 2.3.3

For (2.14),

$$\begin{aligned}
 (Lp)_i &= \sum \delta_{ij} w_{ki} P_i - G_i \sum_{k \in N_i} a_k w_{ki} + \sum_{k \in N_i} b_k w_{ki} \\
 &= \sum \delta_{ij} w_{ki} P_i - G_i \sum_{k \in N_i} a_k w_{ki} k_0 a_k w_{ki} \\
 &= \sum \delta_{ij} w_{ki} P_i - G_i \sum_{k \in N_i} (C_k (\sum_{j \in N_k} w_{kj} G_j P_j - k \overline{p_k})) w_{ki} k_0 a_k w_{ki} \\
 &= \sum_j (\sum_{k|i, j \in N_k} (\delta_{ij} w_{ki} - w_{ki} w_{kj} (G_i - k_0) C_k (G_j - k_0))) p_j
 \end{aligned} \tag{2.21}$$

Chapter 3

Hazerd: an outdoor scene dataset and benchmark for Single image dehazing

Introduction

Haze is a common degradation encountered in outdoor images, where image contrast is reduced due to light scattered by particles suspended in the air. Koschmieder [28, 29] proposed a classical physical model to explain haze, in which horizontal airlight from scattering and light reflected by objects, transmitted and attenuated in the propagation through the hazy air, both contribute to the final images and the ratio of their contributions are controlled by the optical thickness of the media between the camera sensor and the object being imaged. The loss of detail caused by haze makes images aesthetically unappealing and also poses challenges for both human and machine vision, making it difficult to recognize and track objects and to navigate.

To mitigate the impact of haze, physics-based algorithms have been proposed for haze removal or *dehazing*. Early works mainly focus on image

dehazing utilizing additional information. Such works includes: in [30], two images under different weather conditions for the same location are exploited for haze removal; in [31], rough geological information is used; and in [32], the fusion of near-infrared images and the hazy images are utilized for enhancing the details. These algorithms need multiple images or additional information. Though proved to be efficient, these works need either multiple images or information other than RGB images, which may be inaccessible for general applications, for example outdoor image dehazing taken by mainstream consumer-level digital cameras.

In this chapter we focus on single image dehazing methods that typically seek to estimate both the airlight and the transmission from a single hazy image, which is an ill-posed problem. To address the ill-posedness most algorithms impose additional constraints or assumptions to obtain solutions. In [33, 34, 9], color-line or haze-line has been used for modeling the spatial variance of similar color objects. In [35], constraints on air veil are imposed based on the physical model. In [36], the assumption of higher contrast and local smoothness are introduced. In [6], a dark channel prior is proposed that postulates that the color channel with lowest intensity represents airlight. The dark prior is extended in [7] to accommodate color boundary constraints. An alternative color attenuation prior is used with supervised learning in [8] and image fusion based approaches are proposed in [37, 38]. In [39], a Bayesian framework for haze estimation is described. In [10, 11], deep learning networks are designed for estimating the transmission map.

Despite the large number of algorithms proposed for single image dehazing, there are no established criteria or metrics for their evaluation and past publications have primarily relied on subjective comparisons on a limited number of images, with different publications using different sets of images. Three datasets: FRIDA [40], D-hazy [5], and CHIC [41] have been proposed in prior work for objective evaluation of algorithms. FRIDA [40] is rather specialized and provides several synthetic hazy road images from a driver’s viewpoint. D-hazy uses depth images from the Middlebury [2] and the NYU depth V2 [42] which are indoor scenes not representative of the typical dehazing applications. CHIC uses a fog machine in an indoor environment and provides 2 indoor scenes with known objects (e.g. Macbeth color checker) and 2 scenes that include outdoor content seen through windows. The indoor fog generation makes these images atypical, particularly for the case where an outdoor haze-free scene is seen from haze within the room.

In this chapter, we provide a new image dataset *HazeRD*, *Haze Realistic Dataset*, for benchmarking of dehazing algorithms that consists of ten different actual outdoor scenes at high resolution with simulated haze under five different weather conditions, where realistic parameter values are chosen based on scattering theory. As compared to the indoor scenes used in [5], these scenes are more representative of the outdoor conditions under which dehazing is of interest and they correspond to actual images as opposed to the synthetically generated versions in [40]. We benchmark a number of single image dehazing algorithms both on the proposed new dataset and on the ex-

isting D-hazy dataset [5]. Our results demonstrate that there are significant differences between the performance of the different algorithms on different datasets and the rank order of algorithms is by no means constant over the different datasets, thereby emphasizing the need for datasets like ours that are matched with realistic conditions under which dehazing is utilized.

The chapter is organized as follows. Section 3.2 discusses the physical model for atmosphere scattering and image formation; Section 3.3 describes the simulation process; Section 3.4 discusses the HazeRD dataset; Section 3.5 briefly summarizes some state of art dehazing techniques and their performance; at last, Section 3.6, discusses some noteworthy points about dehazing problems and the future work that may be done with HazeRD.

physical model for haze

Atmospheric optics interprets the propagation of light in haze or fog as the process of light random diffusion and reflection by the particles in the air. Hazy weather is caused by the particles, mainly water droplets distributed in the air, which vary in both the size and the density. Haze, fog and cloud in principle share the same components, namely the air with multiple sized water droplets, but differ in the average size and distribution. The scattering of single water droplet has different behaviours regarding to the size, or the radius of the droplet: Rayleigh scattering deals with conditions that the radius is of the size of visible light wavelengths, which corresponding to almost

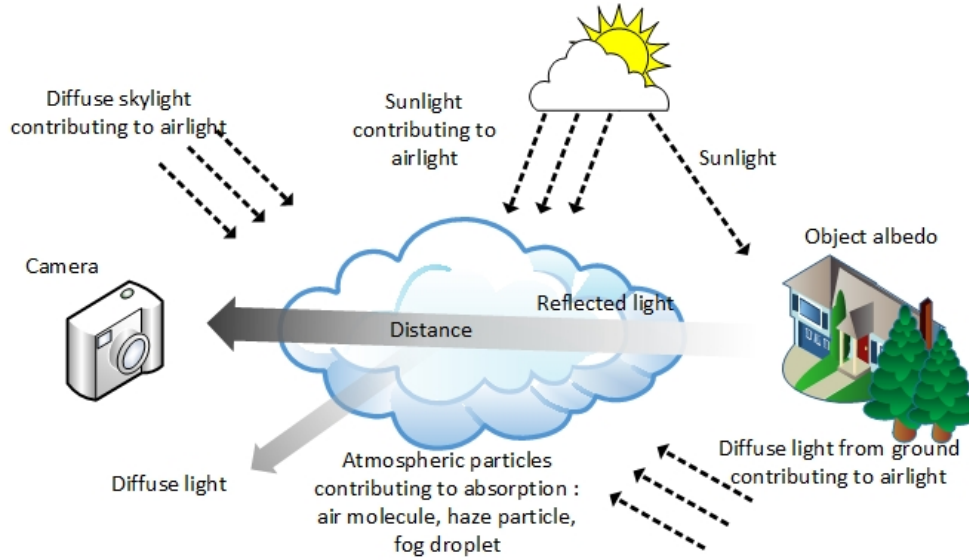


Figure 3.1: The physical model for haze. Light at the sensor is composed of airlight scattered by particles suspended in the air and light reflected from imaged objects, which is attenuated when propagating through the air.

clear weather; and Mie scattering deals with those with the radius far beyond the wavelengths, which displays high anisotropy in the scattered light. The accumulated contribution of the distribution of the water droplets ultimately form the scattered light of the incident light. Table 3.1 lists different weather and the corresponding particles making the main contribution.

As refer to above, Mie scattering theory [29, Chap. 5, 6], which applies when particle sizes are significantly larger that the wavelengths of light involved, can be used to analyze light propagation under hazy conditions. While the exact details of Mie theory are quite involved, image formation under hazy weather can be modeled by taking two main factors into account: the airlight and the attenuation. The physical scenario is depicted

Type	Radius(μm)	Concentration(cm^{-3})
Air molecule	10^{-4}	10^{19}
Aitken nucleus	10^{-4} - 10^{-2}	10^4 - 10^2
Haze Particle	10^{-2} -1	10^2 -1
Fog droplet	1-10	100-10
Cloud droplet	1-10	300-10
Raindrop	10^2 - 10^4	10^{-2} - 10^{-5}

Table 3.1: Particle sizes for different weather

in Fig. 3.1. Due to the scattering of light by the particles in the haze, light from objects attenuates as it propagates from the object to the camera with an intensity that declines exponentially with distance. At the same time, part of the ambient illumination is scattered by the haze particles into the camera as *airlight* that increases the intensity of the image. Assuming a homogeneous haze and a uniform ambient illumination, the spectral irradiance incident on the camera sensor plane from an object with spatially uniform spectral irradiance E_λ can be written as [28],

$$I_\lambda = t_\lambda E_\lambda + (1 - t_\lambda) A_\lambda, \quad (3.1)$$

where λ denotes the wavelength, A_λ is the *airlight*, and $t_\lambda = e^{-d\beta_\lambda}$ is the so called *transmission*, with β_λ denoting the scattering coefficient for the haze particles, and d denoting the distance between the object and the camera. The product $d\beta_\lambda$ is called the *optical thickness*. Observe that as the distance d increases, the contribution of airlight increases while the light from the object diminishes, leading to reduced contrast. The image captured by a

typical three channel RGB (red-green-blue) camera can then be expressed as,

$$I_C = \int_{\lambda} t_{\lambda} E_{\lambda} R_{\lambda,C} + (1 - t_{\lambda}) A_{\lambda} R_{\lambda,C} d\lambda, \quad (3.2)$$

where $C \in \{R, G, B\}$ represents the image channel and $R_{\lambda,C}$ the camera spectral response of the channel.

Haze simulation

In dense haze or fog, the scattering coefficient β_{λ} is almost constant over the visible spectral region, and therefore we can simplify (3.2) by setting $\beta_{\lambda} = \beta$ for all wavelengths λ . The captured image channel intensities can then be represented as

$$I_C(x, y) = E_C(x, y)t(x, y) + A_C(x, y)(1 - t(x, y)), \quad (3.3)$$

$$t(x, y) = e^{-\beta d(x, y)},$$

where (x, y) denotes spatial location,

$$E_C(x, y) = \int_{\lambda} E_{\lambda}(x, y) R_{\lambda,C} d\lambda,$$

is the irradiance of the object received by camera sensor in the absence of haze, i.e., the haze-free image, and $A_C = \int_{\lambda} A_{\lambda} R_{\lambda,C} d\lambda$ is the airlight, and the depth $d(x, y)$ denotes the distance of the object imaged at (x, y) from

the camera plane. Note that a key advantage of the simplified model is that hazy images can be simulated using only haze free images along with depth information using the scattering parameter β and the airlight A_C and the spectral distributions on the right hand side of (3.2), which are invariably unavailable, are not required for the haze simulation. The color and gamma correction [43] in encoding the raw camera sensor values into digital images, however, need to be accounted for. Images are typically encoded in the sRGB color space [44]. We assume that the color correction matrix is absorbed into the channel sensitivities in (3.2) and the "linear" intensity values from (3.3) are nonlinearly encoded into the digital image values via the transform specified in the sRGB standard [44], viz.,

$$C = \begin{cases} 12.92C_L & C \leq 0.0031308, \\ 1.055C_L^{1/2.4} - 0.055 & C > 0.0031308, \end{cases} \quad (3.4)$$

here C_L is the linear RGB value for each channel, C is the corresponding sRGB encoded value, where the transformation is specified on a $[0 - 1]$ range which is then mapped to the 8-bit digital encoding. For the simulation, the inverse of the transformation in (3.4) is applied to the recorded haze free image after mapping the data into the $[0 - 1]$ range and once the simulated hazy images are obtained via (3.3) the transformation in (3.4) is applied and the images are re-encoded as 8-bit representations.

The scattering parameter β depends on the weather conditions. Its value

is specified in terms of the intuitive notion of visual range [29, pp. 42], which is defined, under daylight conditions, as the distance at which the apparent contrast between a dark object and the horizon sky becomes equal to the just noticeable contrast threshold ϵ for an observer (usually set to 0.02). Specifically, the scattering parameter is obtained from the visible range R_m via the relation $\beta = -\ln(\epsilon)/R_m$. HazeRD simulates five different conditions from light to dense fog, for which the visible range and the scattering parameter are listed in Table. 3.2. For simulating hazy images, HazeRD uses (3.3) with these parameter values along with captured haze free images that also have an associated depth map $d(x, y)$ available. Fig. 3.2 summarizes the haze simulation process.

	50m	100m	200m	500m	1000m
weather condition	dense	thick	thick	moderate	light
scattering coef. β	78.2	39.1	19.6	7.82	3.91

Table 3.2: The visual range in HazeRD, and the corresponding weather condition and the scattering coefficient β .

dataset

For performance of single image dehazing algorithms, HazeRD provides ten scenes, each one having a clean RGB image and a ground truth depth map. The dataset is derived from the architectural biometrics project [45, 46] on which we first estimate the dense depth maps for each scene by fusing structure from motion and lidar [12]. Five different weather conditions are

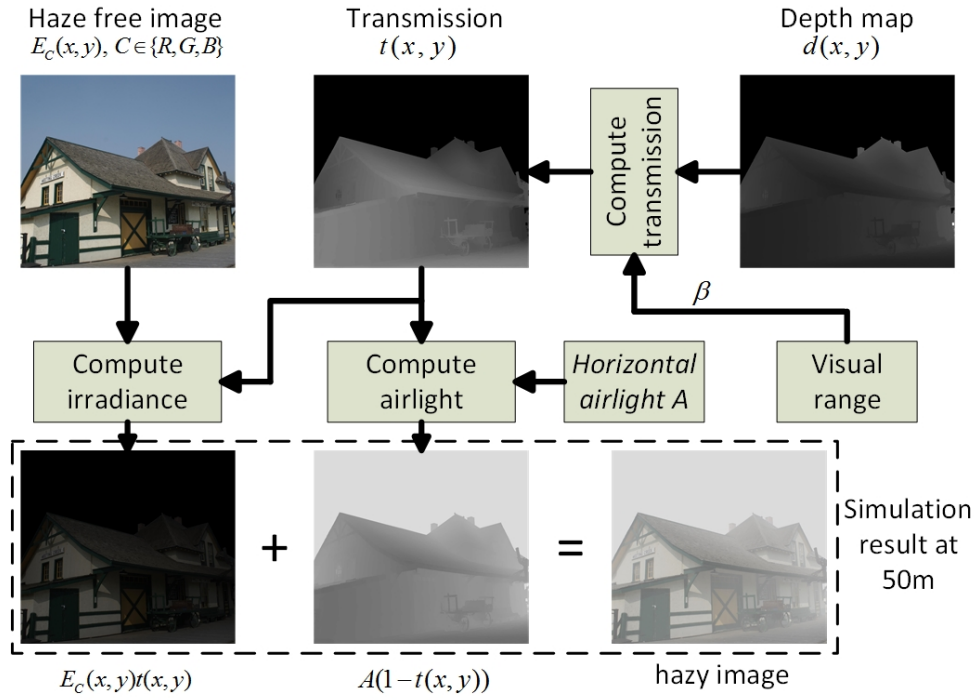


Figure 3.2: Flow for haze simulation based on (3.3).

simulated in each scene, ranging from light fog to dense fog, in order to test the robustness of dehazing algorithms. See Fig. 3.3 for the dataset generation and benchmark block diagram. The simulation of haze and fog is performed by inverse application of (3.4) followed by (3.3), re-white balance and redo the gamma correction. The color image values are converted to $[0, 1]$ for implementing the color space transferring. Then, we use a weighted median filter [47] and the triangular interpolation for the refinement of the depth map. The airlight is set to 0.86, to ensure a visual vividness of objects in overcast weather. Sky area, where typically depth values are missing, is set to have the distance of two times the visual range, which ensures that the transmission is of the order 10^{-4} and that almost no noise is introduced for normal pictures. A sample of hazy images derived from one of the original images in HazeRD is shown in Fig. 3.4 for a number of different visual ranges. Figure. 3.5 shows the depth histograms of HazeRD and for the Middlebury and NYU datasets that also provide images and depth maps required for haze simulation. Compared to the other two datasets which focus primarily on indoor images, the outdoor images in HazeRD span a much larger range of distance ranges and show clear clustering of different object depths. For each scene and the weather condition, a noisy version is provided. Considering the complexity of the real environment, strictly homogeneity is impossible. In atmospheric optics the main component of random fluctuations usually can be expressed by low-order Seidel aberration; here we use the Perlin noise to simulate this phenomenon [40].

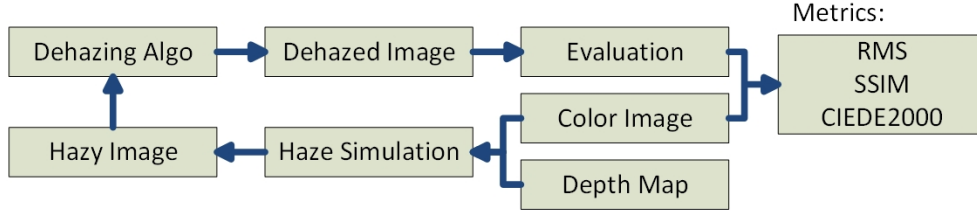


Figure 3.3: Benchmarking workflow for evaluating dehazing algorithms using the HazeRD(proposed) and D-Hazy [5] dataset.



Figure 3.4: HazeRD Samples. From left to right: a hazy image, with the visual range of 50m, 100m, 200m, and 500m respectively.

benchmark: algorithms and datasets

As mentioned above, a typical dehazing algorithm usually has two stages: first, some priors or constraints are formulated to regularize the underconstrained problem, then a loss function is minimized to determine a solution; which is then refined using the hazy images, mainly to eliminate halos or artificial details. In this section, we benchmark six state of art dehazing algorithms described in the following, which we referred to by the labels corresponding to the first author’s last name as: (1) He [6], (2) Meng [7], (3) Zhu [8], (4) Berman [9], (5) Cai [10] and (6) Ren [11]. We briefly summarize the algorithms in order to subsequently understand their success and failure. Some examples of these techniques on HazeRD and synthesized NYU and

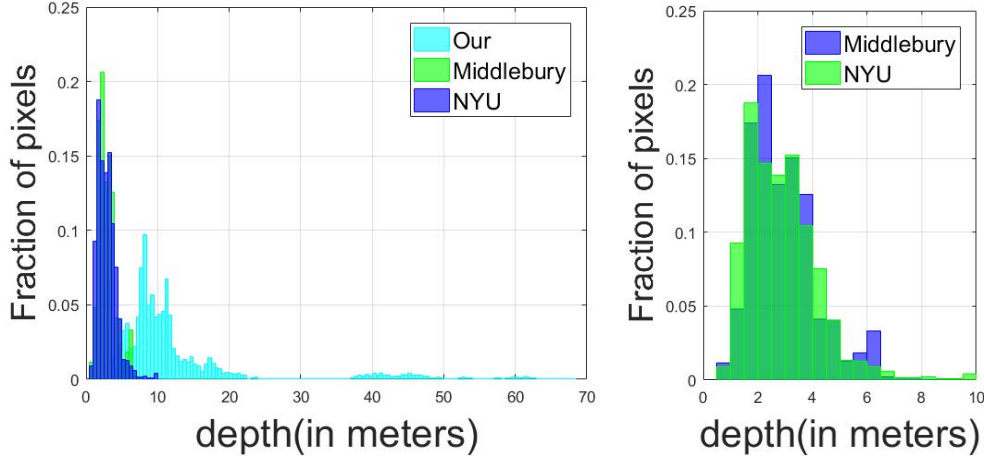


Figure 3.5: Histograms of 3 dataset. Left: HazeRD, Middlebury, and NYU, right: the Middlebury dataset and NYU.

Middlebury dataset are showed in Fig. 3.6, Fig. 3.7, Fig. 3.9, and Fig. 3.8.

He [6] observed that in haze free images, usually the lowest value of a pixel among three channels is close to zero. Thereby from (3.3) in hazy images, the lowest value, called dark channel prior, is an approximation of the transmission. Soft-matting or a guided filter is used as refinement to fit the estimated transmission to the object outlines. This prior is developed further by others, for example into Meng’s [7] color boundary prior and Zhu’s [8] color attenuation prior. The color boundary prior argues that for each image, the color directions are constrained in a cube. The dark channel prior can be derived from the boundary prior with appropriate choice of parameters. The color attenuation prior assumes that the depth can be modeled based on on pixels’ saturation and intensity.

Fattal and Berman [34, 9] developed single image dehazing algorithms

from the color consistency view, called color-line, or haze-line. In their work, the colors of pixels are assumed to be consistent in a small patch of the object. Given the image formation process, patches of the same color should be co-linear, resulting in the so called color-line, and the shifts indicate the optical distance. The difficulty lies in detecting validated patches and in interpolation for the invalid ones. Haze-line is the clustering of the quantized colors, which avoid the complication of patch detection.

Besides these algorithms with strong assumptions, deep learning concepts are also exploited in dehazing algorithms. Cai [10] proposed a four-layer network consisting of a CNN layer, a multi-scale mapping layer, a max pooling layer and a fully-connected layer. The training set is formed from synthesized patches with homogeneous transmission. Ren [11] proposed a coarse-to-fine network consisting of cascade of CNN layers. The training set is obtained as crops from the NYU dataset. Both methods trained the neural network to compute the transmission.

The performance of dehazing techniques can be evaluated from two perspectives: the accuracy of estimated transmission maps and the fidelity of the dehazed images, each with respect to the corresponding ground truth. We use root mean square (RMS) error to evaluate the difference of the estimated transmission and the ground truth, SSIM [48] to evaluate the similarity of dehazed images and the original haze free images, and CIEDE2000 [49] to evaluate the color fidelity. Results for the algorithms benchmarked and across the different individual datasets are summarized in Table. 3.3. These results

show that typically the transmission values (which are always smaller than 1) have a large error, and the SSIM and CIEDE2000 metrics also show that the dehazed images have significant perceptible difference with the original images. The performance of most of the techniques tested here varies with different weather conditions. Table. 3.3 also lists the weather condition, or equivalently visual range, that yields the best average performance for each algorithm. The tabulated values indicate that algorithms based on priors are limited largely due to the scene and the weather. Generally, these algorithms are not reliable in the sense of dehazing. The visual range for indoor dataset, Middlebury and NYU dataset, is expanded to 5m, 10m, 20m and 50m compared to [5]. Contrasting the differences between indoor and outdoor scenes is important for that generally dehazing techniques are more likely to be used in the latter case. For each algorithm, we run the T-test on the RMS, the SSIM and the CIEDE2000 of HazeRD and the two reference datasets. These datasets shows statistically significant differences between the performances of most algorithms ($\alpha = 5\%$). This demonstrates the value of HazeRD as an alternative benchmark for dehazing algorithms: the results on the indoor datasets with limited depth range do not appear to hold for the outdoor datasets.



(a)



(b)



(c)



(d)



(e)



(f)



(g)

Figure 3.6: Example of algorithms' performances. (a): a haze free image from HazeRD, and the results of dehazing of a corresponding hazy image obtained with: (b):He [6], (c): Meng [7], (d): Zhu [8], (e): Berman [9], (f): Cai [10], and (g): Ren [11].

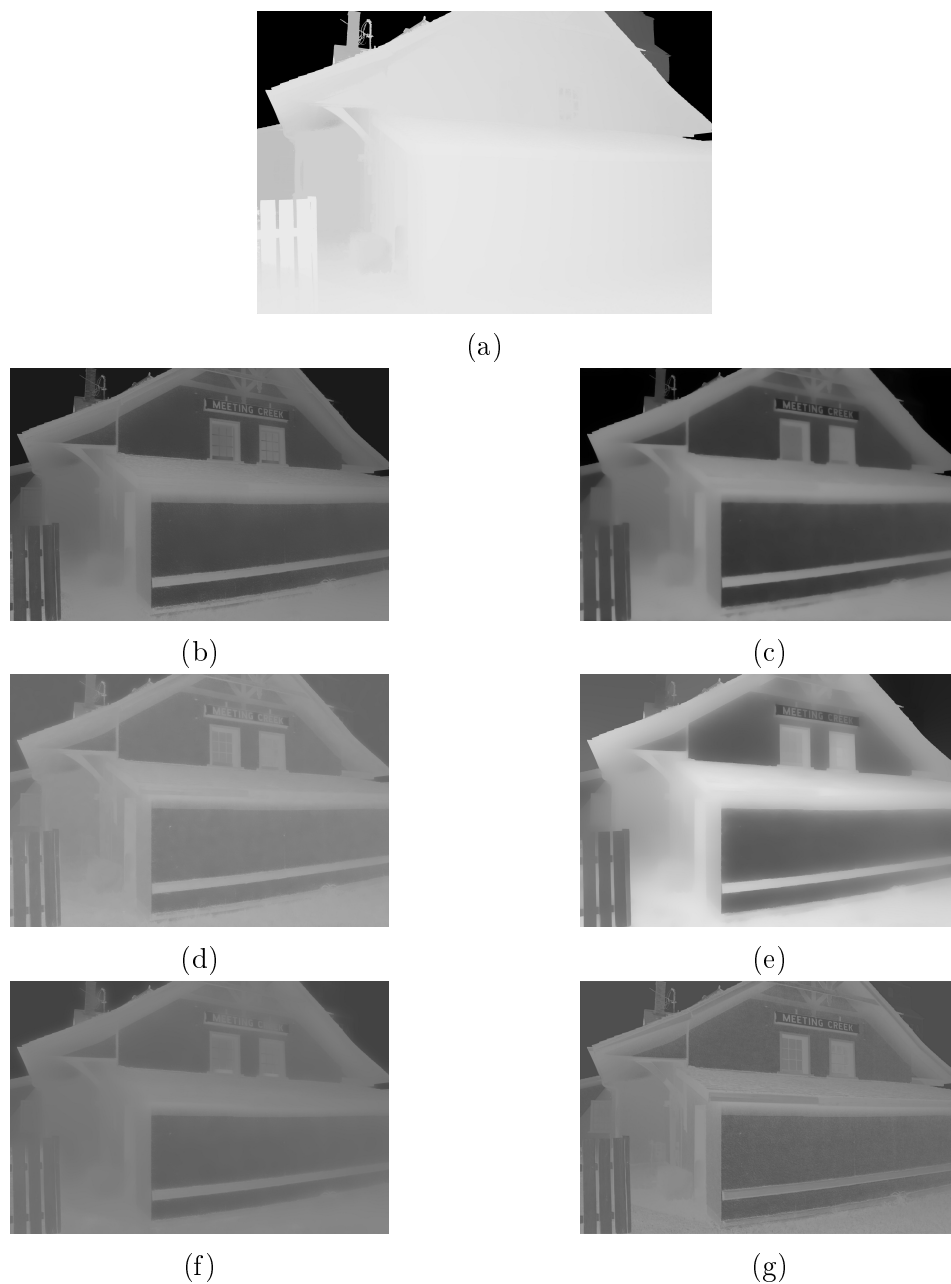
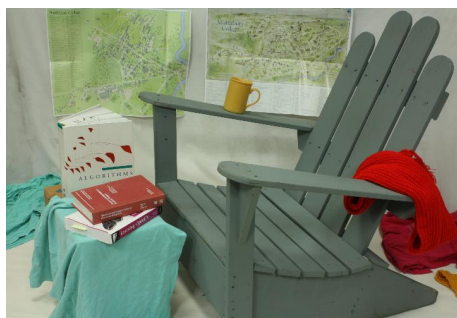
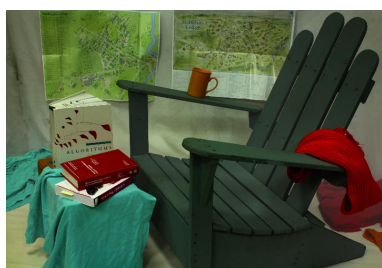


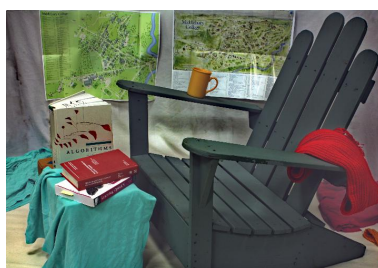
Figure 3.7: Example of algorithms' performances. (a): the ground truth transmission for the haze free image in 3.6, and the transmission estimates corresponding to the estimated images obtained with: (b):He [6], (c): Meng [7], (d): Zhu [8], (e): Berman [9], (f): Cai [10], and (g): Ren [11].



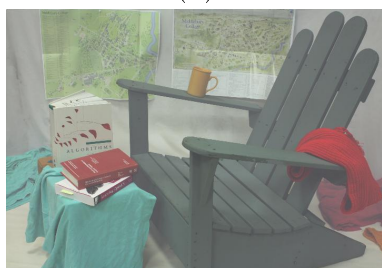
(a)



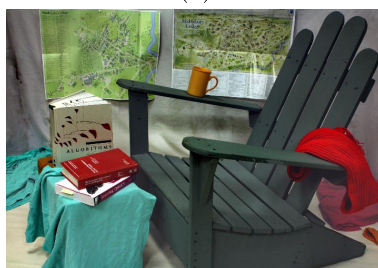
(b)



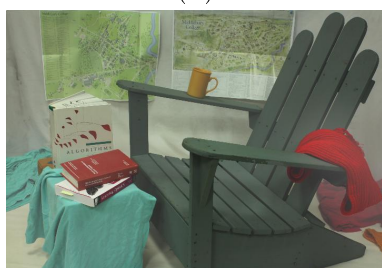
(c)



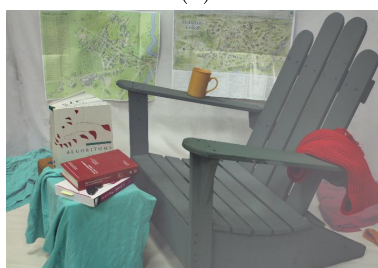
(d)



(e)



(f)



(g)

Figure 3.8: Example of algorithms' performances. (a): a haze free image from Middlebury dataset, and the results of dehazing of a corresponding hazy image obtained with: (b):He [6], (c): Meng [7], (d): Zhu [8], (e): Berman [9], (f): Cai [10], and (g): Ren [11].



(a)



(b)



(c)



(d)



(e)



(f)



(g)

Figure 3.9: Example of algorithms' performances. (a): a haze free image from NYU dataset, and the results of dehazing of a corresponding hazy image obtained with: (b):He [6], (c): Meng [7], (d): Zhu [8], (e): Berman [9], (f): Cai [10], and (g): Ren [11].

		He [6]	Meng [7]	Zhu [8]	Berman [9]	Cai [10]	Ren [11]
HazeRD	RMS	0.2978 \pm 0.0902	0.3007 \pm 0.0927	0.1991 \pm 0.0641	0.2193 \pm 0.1113	0.2765 \pm 0.1002	0.2680 \pm 0.0990
	SSIM	0.5148 \pm 0.1293	0.6464 \pm 0.1723	0.6692 \pm 0.1979	0.5962 \pm 0.1593	0.4770 \pm 0.1808	0.6664 \pm 0.2013
	CIEDE2000	20.7038 \pm 4.2821	17.2284 \pm 5.0159	15.4259 \pm 5.7246	19.0647 \pm 4.8029	19.5447 \pm 4.8312	14.5432 \pm 6.0696
	Best at	dense	thick to light	moderate to light	no preference	dense	light
Middlebury	RMS	0.2142 \pm 0.1242	0.2461 \pm 0.1408	0.1921 \pm 0.0985*	0.2551 \pm 0.1084*	0.1792 \pm 0.0792	0.2022 \pm 0.0980
	SSIM	0.7046 \pm 0.1383	0.5788 \pm 0.2487	0.6394 \pm 0.2524	0.6093 \pm 0.2200	0.6227 \pm 0.2556	0.5978 \pm 0.2522
	CIEDE2000	19.3802 \pm 6.6979	18.4290 \pm 4.7335*	19.8333 \pm 12.2146*	19.1038 \pm 6.6742*	18.8720 \pm 10.4724*	20.0272 \pm 11.1873
	Best at	10m/20m	50m	50m	50m	20m/50m	50m
NYU	RMS	0.2074 \pm 0.1121	0.2404 \pm 0.1228	0.1998 \pm 0.0845*	0.2119 \pm 0.0769*	0.1976 \pm 0.0772	0.1995 \pm 0.0758
	SSIM	0.6478 \pm 0.0518	0.7203 \pm 0.0596	0.7029 \pm 0.1668*	0.7100 \pm 0.0763	0.6773 \pm 0.1261	0.7044 \pm 0.1459*
	CIEDE2000	19.7528 \pm 3.6458*	16.9486 \pm 2.8969*	16.2086 \pm 8.1587*	15.7918 \pm 2.8642	16.5856 \pm 5.5898	15.4301 \pm 7.3678*
	Best at	10m	10m 50m	50m	20m/50m	20m	50m

Table 3.3: Performance of different dehazing methods on HazeRD, Middlebury, and NYU datasets. Each numerical entry is represented as the average over the images in the dataset \pm the standard deviation. The best performing algorithm for each dataset is indicated in bold font, and * in the Middlebury and NYU datasets indicates cases where the difference with respect to HazeRD was not statistically significant (5% level).

The regions in which these algorithms fail on the HazeRD database also provides insight. Algorithms based on dark channel assume all white or bright area is mainly caused by skylight. In HazeRD, there are several white or bright walls which undermine these assumptions, for example, see figure. 3.10. Typical dark channel values are above 0.2, and the error is almost linear in all weather conditions. The haze-line algorithm also has difficulty on bright surfaces, especially rough surfaces. The fluctuations in such surfaces are amplified to a large color difference. Cai's [10] algorithm tends to underestimate the transmission in sky area, which may be caused by the training set, generated by cropping small patches merely from images, and assigning a uniform random depth for each patch.

discussion

As we have seen from the experiment results, none of the algorithms benchmarked here provides a sound estimate for the transmission on the HazeRD dataset and each algorithm suffers from some artifacts or color infidelity. In other words, most algorithms seem to focus on enhancing the contrast or saturation without regard to what the true haze free image would be. Partly the reason is that some priors don't hold true on these images. Generally most priors are observations based on particular kinds of images, not the physical model itself. Another easily overlooked problem is that the skylight is not actually uniform but exhibits a gradual variation. In fact, we expect

the clear dehazed sky to be blue and not the gray or white that is commonly observed. Last but not least, the performance of most of the techniques tested here varies depending on the weather condition, which indicates that proposed priors should be tested systematically and that training sets for deep learning methods should include more comprehensive situations.

The results we obtained here indicate that indoor and outdoor scenes have some fundamental differences on dehazing techniques and HazeRD provides a valuable alternative for the benchmarking for dehazing algorithms in more realistic outdoor settings. HazeRD is also a potential dataset for benchmarking of other algorithms including monocular image depth estimation, outdoor scene segmentation, and other applications requiring RGB-D information.

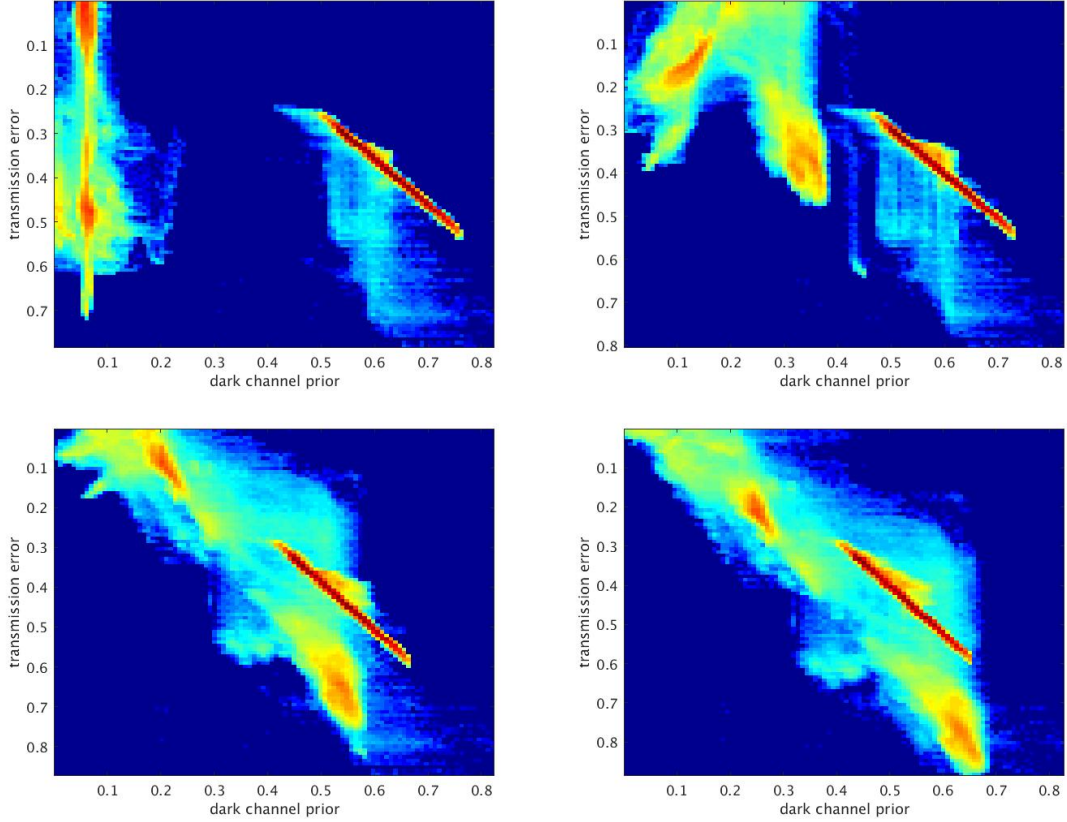


Figure 3.10: Scatter diagram of one scene of the dark channel prior error (y axis) and the transmission error (x axis) in log scale. Each point represents the density of pixels with similar prior-transmission ratios. See the bright line. The dark prior channel is above 0.2 in most area, which is contradictory to the assumption. The transmission error is almost linearly related to the dark channel error. From left to right, fog: dense, thick, moderate, light.

Chapter 4

Conclusion

This thesis has two main contributions. First, an accurate depth map upsampling algorithm is proposed. This method is based on local linear fitting and utilizes the color information to improve the edge quality. We also propose a memory saving implementation to solve the optimization problem. Second, we using RGB-D images to construct an outdoor scene dataset, *HazeRD*, and the benchmarking for dehazing algorithms. Compared to existing dataset, *HazeRD* provides outdoor scenes instead of indoor scenes, which is the typical conditions for the application of dehazing algorithm, and ground truth of high accuracy.

The work in this thesis also has some limitations. The proposed depth upsampling algorithm is extremely time consuming compared to naive methods such as bilinear or bicubic interpolation. In future work, one direction is to use parallel techniques to accelerate the computing. Besides, the benchmarking results of state of art dehazing algorithms indicates that the main bottleneck lies in the consistency of the estimated transmission of the same

objects. In future work, new dehazing algorithms combining image segmentation algorithms should be considered.

Bibliography

- [1] D. Scharstein and R. Szeliski, “High-accuracy stereo depth maps using structured light,” in *IEEE Intl. Conf. Comp. Vision, and Pattern Recog.*, vol. 1, 2003, pp. I–195.
- [2] D. Scharstein, H. Hirschmüller, Y. Kitajima, G. Krathwohl, N. Nešić, X. Wang, and P. Westling, “High-resolution stereo datasets with subpixel-accurate ground truth,” in *German Conf. on Pattern Recog.*, 2014, pp. 31–42.
- [3] J. Yang, X. Ye, K. Li, C. Hou, and Y. Wang, “Color-guided depth recovery from RGB-D data using an adaptive autoregressive model,” in *IEEE Trans. Image Proc.*, vol. 23, no. 8, 2014, pp. 3443–3458.
- [4] D. Ferstl, C. Reinbacher, R. Ranftl, M. Rütger, and H. Bischof, “Image guided depth upsampling using anisotropic total generalized variation,” in *IEEE Intl. Conf. Comp. Vision.*, 2013, pp. 993–1000.
- [5] C. Ancuti, C. O. Ancuti, and C. D. Vleeschouwer, “D-hazy: A dataset to evaluate quantitatively dehazing algorithms,” in *IEEE Intl. Conf. Image Proc.*, Sept 2016, pp. 2226–2230.
- [6] K. He, J. Sun, and X. Tang, “Single image haze removal using dark channel prior,” *IEEE Trans. Pattern Anal. Mach. Intel.*, vol. 33, no. 12, pp. 2341–2353, 2011.
- [7] G. Meng, Y. Wang, J. Duan, S. Xiang, and C. Pan, “Efficient image dehazing with boundary constraint and contextual regularization,” in *IEEE Intl. Conf. Comp. Vision.*, 2013, pp. 617–624.

- [8] Q. Zhu, J. Mai, and L. Shao, “A fast single image haze removal algorithm using color attenuation prior,” *IEEE Trans. Image Proc.*, vol. 24, no. 11, pp. 3522–3533, 2015.
- [9] D. Berman, T. Treibitz, and S. Avidan, “Non-local image dehazing,” in *IEEE Intl. Conf. Comp. Vision, and Pattern Recog.*, June 2016.
- [10] B. Cai, X. Xu, K. Jia, C. Qing, and D. Tao, “Dehazenet: An end-to-end system for single image haze removal,” *IEEE Trans. Image Proc.*, vol. PP, no. 99, pp. 1–1, 2016.
- [11] W. Ren, S. Liu, H. Zhang, J. Pan, X. Cao, and M.-H. Yang, “Single image dehazing via multi-scale convolutional neural networks,” in *Proc. European Conf. Computer Vision*, 2016.
- [12] L. Ding and G. Sharma, “Fusing structure from motion and lidar for accurate dense depth map estimation,” 2017, submitted to IEEE ICASSP 2017.
- [13] J. Diebel and S. Thrun, “An application of Markov random fields to range sensing,” in *Adv. in Neural Info. Proc. Sys.*, vol. 5, 2005, pp. 291–298.
- [14] Q. Yang, R. Yang, J. Davis, and D. Nistér, “Spatial-depth super resolution for range images,” in *IEEE Intl. Conf. Comp. Vision, and Pattern Recog.*, 2007, pp. 1–8.
- [15] K. He, J. Sun, and X. Tang, “Guided image filtering,” in *Proc. European Conf. Computer Vision*, 2010, pp. 1–14.
- [16] J. Kopf, M. F. Cohen, D. Lischinski, and M. Uyttendaele, “Joint bilateral upsampling,” in *ACM Trans. on Graphics*, vol. 26, no. 3, 2007, p. 96.
- [17] J. Park, H. Kim, Y.-W. Tai, M. S. Brown, and I. Kweon, “High quality depth map upsampling for 3D-TOF cameras,” in *IEEE Intl. Conf. Comp. Vision.*, 2011, pp. 1623–1630.
- [18] J. Lu, D. Min, R. S. Pahwa, and M. N. Do, “A revisit to MRF-based depth map super-resolution and enhancement,” in *IEEE Intl. Conf. Acoust., Speech, and Signal Proc.*, 2011, pp. 985–988.

- [19] A. Harrison and P. Newman, “Image and sparse laser fusion for dense scene reconstruction,” in *Field and Service Robotics*, 2010, pp. 219–228.
- [20] S. Krishnamurthy and K. R. Ramakrishnan, “Image-guided depth map upsampling using normalized cuts-based segmentation and smoothness priors,” in *IEEE Intl. Conf. Image Proc.*, 2016, pp. 554–558.
- [21] K. Uruma, K. Konishi, T. Takahashi, and T. Furukawa, “High resolution depth image recovery algorithm based on the modeling of the sum of an average distance image and a surface image,” in *IEEE Intl. Conf. Image Proc.*, 2016, pp. 2836–2840.
- [22] A. Levin, D. Lischinski, and Y. Weiss, “A closed-form solution to natural image matting,” in *IEEE Trans. Pattern Anal. Mach. Intel.*, vol. 30, no. 2, 2008, pp. 228–242.
- [23] R. Merris, “Laplacian matrices of graphs: a survey,” *Linear algebra and its applications*, vol. 197, pp. 143–176, 1994.
- [24] K. He, J. Sun, and X. Tang, “Fast matting using large kernel matting laplacian matrices,” in *IEEE Intl. Conf. Comp. Vision, and Pattern Recog.* IEEE, 2010, pp. 2165–2172.
- [25] C. Yu, G. Sharma, and H. Aly, “Computational efficiency improvements for image colorization,” in *IS&T/SPIE Electronic Imaging*, 2014, pp. 902 004–902 004.
- [26] D. Scharstein and C. Pal, “Learning conditional random fields for stereo,” in *IEEE Intl. Conf. Comp. Vision, and Pattern Recog.*, 2007, pp. 1–8.
- [27] H. Hirschmuller and D. Scharstein, “Evaluation of cost functions for stereo matching,” in *IEEE Intl. Conf. Comp. Vision, and Pattern Recog.*, 2007, pp. 1–8.
- [28] H. Koschmieder, “Theorie der horizontalen sichtweite,” in *Beitrage zur Physik der freien Atmosphere*, 1924.
- [29] E.J.McCartney, “Optics of the atmosphere: Scattering by molecules and particles.” New York, NY: John Wiley and Sons, 1976.

- [30] S. G. Narasimhan and S. K. Nayar, "Contrast restoration of weather degraded images," *IEEE Trans. Pattern Anal. Mach. Intel.*, vol. 25, no. 6, pp. 713–724, 2003.
- [31] J. Kopf, B. Neubert, B. Chen, M. Cohen, D. Cohen-Or, O. Deussen, M. Uyttendaele, and D. Lischinski, "Deep photo: Model-based photograph enhancement and viewing," *ACM Trans. on Graphics*, vol. 27, no. 5, p. 116, 2008.
- [32] L. Schaul, C. Fredembach, and S. Süsstrunk, "Color image dehazing using the near-infrared," in *IEEE Intl. Conf. Image Proc.*, 2009, pp. 1629–1632.
- [33] R. Fattal, "Single image dehazing," *ACM Trans. on Graphics*, vol. 27, no. 3, p. 72, 2008.
- [34] —, "Dehazing using color-lines," *ACM Trans. on Graphics*, vol. 34, no. 1, p. 13, 2014.
- [35] J.-P. Tarel and N. Hautiere, "Fast visibility restoration from a single color or gray level image," in *IEEE Intl. Conf. Comp. Vision.*, 2009, pp. 2201–2208.
- [36] R. T. Tan, "Visibility in bad weather from a single image," in *IEEE Intl. Conf. Comp. Vision, and Pattern Recog.*, 2008, pp. 1–8.
- [37] C. O. Ancuti, C. Ancuti, C. Hermans, and P. Bekaert, "A fast semi-inverse approach to detect and remove the haze from a single image," in *Asian Conf. on Computer Vision*, 2010, pp. 501–514.
- [38] C. O. Ancuti and C. Ancuti, "Single image dehazing by multi-scale fusion," *IEEE Trans. Image Proc.*, vol. 22, no. 8, pp. 3271–3282, 2013.
- [39] K. Nishino, L. Kratz, and S. Lombardi, "Bayesian defogging," *Intl. J. Computer Vision*, vol. 98, no. 3, pp. 263–278, 2012.
- [40] J.-P. Tarel, N. Hautiere, L. Caraffa, A. Cord, H. Halmaoui, and D. Gruyer, "Vision enhancement in homogeneous and heterogeneous fog," *IEEE Trans. Intell. Transp. Syst.*, vol. 4, no. 2, pp. 6–20, 2012.

- [41] J. El Khoury, J.-B. Thomas, and A. Mansouri, "A color image database for haze model and dehazing methods evaluation," in *International Conference on Image and Signal Processing*. Springer, 2016, pp. 109–117.
- [42] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from rgb-d images," in *Proc. European Conf. Computer Vision*, 2012, pp. 746–760.
- [43] G. Sharma, Ed., *Digital Color Imaging Handbook*. Boca Raton, FL: CRC Press, 2003.
- [44] I. E. Commission *et al.*, "Multimedia systems and equipment—colour measurement and management—part 2-1: Colour management—default rgb colour space—srgb," IEC 61966-2-1, Tech. Rep., 1999.
- [45] L. Ding, A. Elliethy, E. Freedenberg, S. A. Wolf-Johnson, J. Romphf, P. Christensen, and G. Sharma, "Comparative analysis of homologous buildings using range imaging," in *IEEE Intl. Conf. Image Proc.* IEEE, 2016, pp. 4378–4382.
- [46] "Architectural biometrics project website." [Online]. Available: <http://www.architecturalbiometrics.com>
- [47] Z. Ma, K. He, Y. Wei, J. Sun, and E. Wu, "Constant time weighted median filtering for stereo matching and beyond," in *IEEE Intl. Conf. Comp. Vision.*, 2013, pp. 49–56.
- [48] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Trans. Image Proc.*, vol. 13, no. 4, pp. 600–612, April 2004.
- [49] G. Sharma, W. Wu, and E. N. Dalal, "The CIEDE2000 color-difference formula: Implementation notes, supplementary test data, and mathematical observations," *Color Res. & Appl.*, vol. 30, no. 1, pp. 21–30, 2005.