High Quality Depth Refinement with Color Photometric Stereo

Songyou Peng

Supervised by:

Dr. Yvain Quéau Prof. Daniel Cremers

Computer Vision Group

Department of Computer Science

Technical University of Munich

A Thesis Submitted for the Degree of
MSc Erasmus Mundus in Vision and Robotics (VIBOT)

· 2017 ·

Abstract

The abstract will go here....

Research is what I'm doing when I don't know what I'm doing....

Werner von Braun

Contents

List of Figures

List of Tables

Acknowledgments

Leave this part until I finish the whole thesis

Chapter 1

Introduction

1.1 Research Goal

1.2 Outline

1.2.1 References

You can reference other authors by using the *citecommand* [**?**] [**?**]. You are encouraged to use bib files and let bibtex do the job for you.

Chapter 2

Background

Joint estimation of depth, reflectance and illumination for depth refinement

2.1 RGB-D Cameras

2.1.1 General

2.1.2 ASUS Xtion PRO LIVE

2.2 Shape from Shading & Photometric Stereo

2.2.1 Lambertian reflectance model

cite Shape From Shading Emmanuel Prados, Olivier Faugeras https://en.wikipedia.org/wiki/Lambertian_reflectance

https://www.cs.cmu.edu/afs/cs/academic/class/15462-f09/www/lec/lec8.pdf http://www.cs.virginia.edu/~gfx/Courses/2011/ComputerVision/slides/lecture20_pstereo.pdf

2.2.2 Surface normal

http://docs.opencv.org/2.4/modules/calib3d/doc/camera_calibration_and_3d_reconstruction.html

orthographic model perspective model

2.3 Intrinsic Image Decomposition

2.4 Depth Map Refinement

2.5 Super-resolution Imaging

Chapter 3

Methodology

Describe the structure of this chapter.

3.1 Pre-Processing

The first step for most of the image processing tasks is to pre-process the initial input image. Due to the hardware limitation of modern inexpensive RGBD sensors, there usually exist holes with missing values on the depth images. Also, the depth data is often noisy so we need to do denoising and acquire a relative smooth surface.

In this section, we will describe respectively the basic depth inpainting and denoising algorithm that we use for our pre-processing.

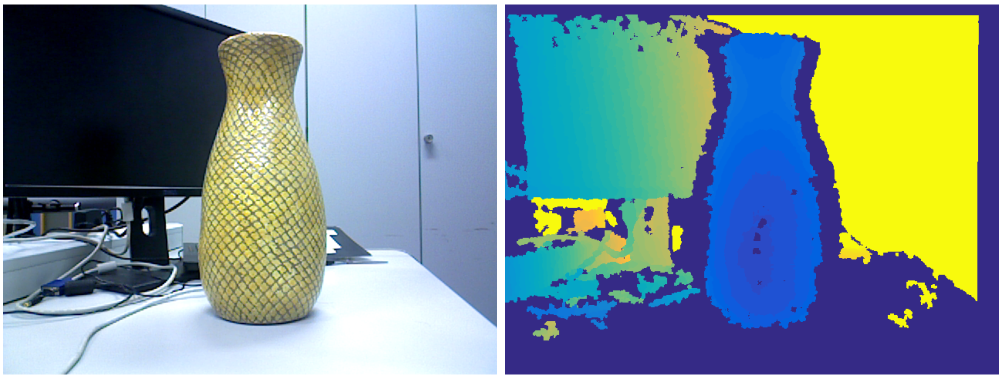
3.1.1 Depth inpainting

Image inpainting itself is a very mutual area and has been widely applied as a useful tool for many modern computer vision applications, e.g, restore the damaged parts of ancient paintings, or remove unwanted texts or objects in a photography [1]. Since the idea of image inpainting is to automatically replace the lost or undesired parts of an image with the neighbouring information by interpolating, we were inspired to apply it to fill in the missing depth information (Fig. 3.1).

It should be noted that, the depth inpainting is applied to the input noisy image so there is no need to use some powerful and advanced algorithms. The only request is to fill the missing areas with inexpensive computational time.

The general mathematical form of a classic inpainting algorithm [1] can be written as follows

$$I^{t+1}(i,j) = I^t(i,j) + \mu U^t(i,j), \forall (i,j) \in \Omega \tag{3.1}$$



(a) RGB image (b) Input depth image

Figure 3.1: The input RGB and depth image of a vase. The brighter color on the image (b), the higher depth values.

where $I(i, j)$ is the pixel value in image I, t is the artificial time step, μ is the updating rate, U is the update information and Ω are the area with missing information.

To build the update map U in each time step, there are two principles that [1] follow. One is the inpainted values inside Ω should be as smooth as possible. The other is the lines reaching the edge of Ω should be continued and cross the missing area, while the values in Ω should be propagated from the nearest neighbours of Ω along the lines.

Again, due to the fact that our input depth images have poor quality, the lines arriving at the boundary $\delta\Omega$ may be incorrect or produced by the noises. Thus, it is reasonable that our initial depth inpainting problem focuses on the smooth propagation from the neighbours and fill in the holes.

In each pixel (x_0, y_0) inside Ω, U can be modelled as a discrete four-neighbour Laplacian operator:

$$U(x_0, y_0) = \Delta I = 4I(x_0, y_0) - I(x_0 + 1, y_0) - I(x_0 - 1, y_0) - I(x_0, y_0 + 1) - I(x_0, y_0 - 1) \tag{3.2}$$

Now the inpainting problem in Eq. 3.1 can be represented as a minimization problem:

$$\min \iint_{\Omega} |U(x, y)|^2 dxdy \tag{3.3}$$

This problem can be reformulated to a typical linear equation in matrix form:

$$\mathbf{Ax} = \mathbf{b} \tag{3.4}$$

Assuming n is the number of pixel inside Ω and m is the sum of n and the number of neighbouring pixel around the boundary $\delta\Omega$, $\mathbf{A}$ is a $m \times n$ Laplacian matrix, $\mathbf{b}$ is a $m \times 1$ vector containing all the known boundary depth values and the 0 inside Ω. Solving the linear equation with simple least square method, we can acquire the inpainted values. With our this naive image inpainting algorithm, we can fill the holes on the depth image as shown in Fig. 3.2.

3.1.2 Depth denoising

the depth images acquired from the RGB-D cameras with moderate price usually contain various noises. As a standard pre-processing method, the image denoising technique is also applied to our input inpainted depth map. Similar to the state-of-the-art depth refinement methods [2–7], bilateral fitering [8] is used as our depth pre-processing smoother.

The advantages of bilateral filter is reducing the noise while preserving the edge in the input image. More than a regular Gaussian smooth filter, which uses only the difference of the image values (depth in our case) between the center pixel the neighbours, the bilateral filter also utilizes the space difference as a reference to build up the weighting function. The filtered pixel value can be modelled as a weighted sum of neighbouring pixels:

$$\hat{I}(\mathbf{x}) = \frac{1}{W} \sum_{\mathbf{y} \in \mathcal{N}} I(\mathbf{y}) e^{-\left(\frac{\|I(\mathbf{x}) - I(\mathbf{y})\|^2}{2\sigma_r^2} + \frac{\|\mathbf{x} - \mathbf{y}\|^2}{2\sigma_d^2}\right)} \tag{3.5}$$

where $\hat{I}(\mathbf{x})$ is the filtered value at pixel $\mathbf{x}$, $\mathcal{N}$ represents the neighbouring pixels with $\mathbf{x}$ in the center, and W is the the sum of the all the weights. The smoothed result on our input depth image is shown in Fig. 3.2.

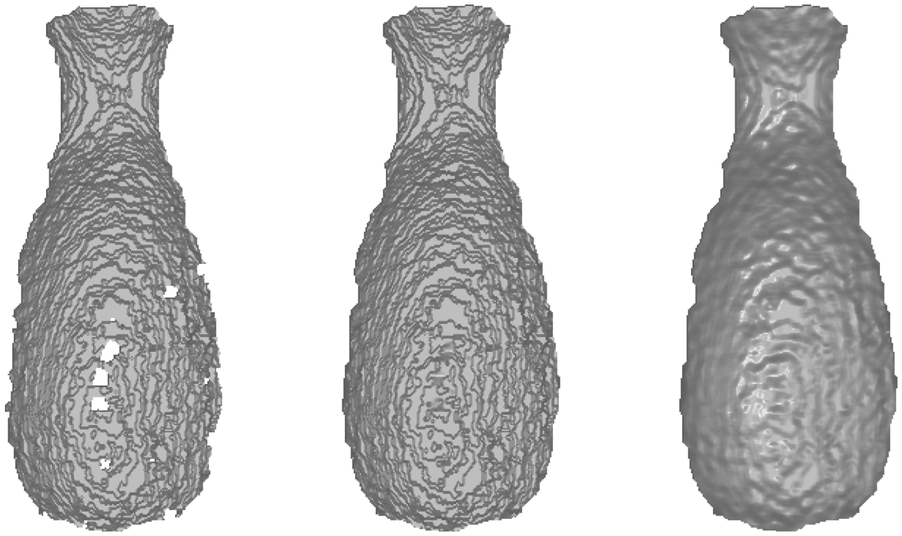
3.2 RGBD-Fusion method

3.2.1 ddd

3.3 Proposed method 1: RGB Ratio Model

3.3.1 Limitations

- *LEDs have to be set up far away from each other.*



(a) Input depth (b) After inpainting (c) After smoothing

Figure 3.2: Illustrations for the pre-processing on the depth of the vase.

- *Natural illumination is a problem.*
- *Only feasible for the simple albedo cases*
- *Non specular objects*

3.4 Proposed method 2: Robust Lighting Variation Model without Regularization

3.4.1 Depth super-resolution

Chapter 4

Results and Evaluation

4.1 RGB-D Cameras

4.1.1 General

4.1.2 ASUS Xtion PRO LIVE

4.2 Shape from Shading & Photometric Stereo

4.3 Intrinsic Image Decomposition

4.4 Super-resolution Imaging

Chapter 5

Conclusion

5.1 RGB-D Cameras

5.1.1 General

5.1.2 ASUS Xtion PRO LIVE

5.2 Shape from Shading & Photometric Stereo

5.3 Intrinsic Image Decomposition

5.4 Super-resolution Imaging

Appendix A

The first appendix

If you need to add any appendix, do it here... Etc.

Bibliography

[1] Marcelo Bertalmio, Guillermo Sapiro, Vincent Caselles, and Coloma Ballester. Image inpainting. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 417–424. ACM Press/Addison-Wesley Publishing Co., 2000.

[2] Qing Zhang, Mao Ye, Ruigang Yang, Yasuyuki Matsushita, Bennett Wilburn, and Huimin Yu. Edge-preserving photometric stereo via depth fusion. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2472–2479. IEEE, 2012.

[3] Roy Or-El, Guy Rosman, Aaron Wetzler, Ron Kimmel, and Alfred M Bruckstein. Rgbd-fusion: Real-time high precision depth recovery. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5407–5416, 2015.

[4] Yudeog Han, Joon-Young Lee, and In So Kweon. High quality shape from a single rgb-d image under uncalibrated natural illumination. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1617–1624, 2013.

[5] Roy Or-El, Rom Hershkovitz, Aaron Wetzler, Guy Rosman, Alfred M Bruckstein, and Ron Kimmel. Real-time depth refinement for specular objects. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4378–4386, 2016.

[6] Mohammadul Haque, Avishek Chatterjee, Venu Madhav Govindu, et al. High quality photometric reconstruction using a depth camera. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2275–2282, 2014.

[7] Lap-Fai Yu, Sai-Kit Yeung, Yu-Wing Tai, and Stephen Lin. Shading-based shape refinement of rgb-d images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1415–1422, 2013.

[8] Carlo Tomasi and Roberto Manduchi. Bilateral filtering for gray and color images. In *Computer Vision, 1998. Sixth International Conference on*, pages 839–846. IEEE, 1998.