Vision with Small Baselines

Chao Liu

CMU-RI-TR-20-50
August, 2020

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Thesis Committee:
Srinivasa G. Narasimhan, co-chair (CMU)
Artur W. Dubrawski, co-chair (CMU)
Aswin C. Sankaranarayanan, (CMU)
Manmohan Chandraker, (UCSD)

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy.

Keywords: Small Baseline Vision. Physics based Vision. Inverse problem. 3D reconstruction. Occlusion. Uncertainty. Diffuse Optical Tomography

Abstract

3D sensing with portable imaging systems is becoming more and more popular in computer vision applications such as autonomous driving, virtual reality, robotics manipulation and surveillance, due to the decreasing expense and size of RGB cameras. Despite the compactness and portability of the small baseline vision systems, it is well-known that the uncertainty in range finding using multiple views and the sensor baselines are inversely related. On the other hand, besides compactness, the small baseline vision system has its unique advantages such as easier correspondence and large overlapping regions across views.

The goal of this thesis is to develop computational methods and small baseline imaging systems for 3D sensing of complex scenes in real world conditions. Our design principle is to physically model the scene complexities and specifically infer the uncertainties for the images captured with small baseline setups.

With this design principle, we make four contributions. In the first contribution, we propose a two-stage near-light photometric stereo method using a small (6 cm diameter) LED ring. The imaging system is compact compared to traditional photometric stereo systems. In the second contribution, we develop an algorithm to simultaneously estimate the occlusion pattern and depth for thin structures from a focal image stack, which is obtained either by varying the focus/aperture of the lens or computed from a one-shot light field image. As the third contribution, we propose a learning-based method to estimate per-pixel depth and its uncertainty continuously from a monocular video stream, with small camera baselines across adjacent frames. These depth probability volumes are accumulated over time as more incoming frames are processed sequentially, which effectively reduces depth uncertainty and improves accuracy, robustness, and temporal stability. Finally, using a pair of high resolution camera and laser projector, we develop a high spatial resolution Diffuse Optical Tomography system that can detect accurate boundaries and relative depth of heterogeneous structures up to a depth of 8mm below a highly scattering medium such as whole milk.

We showcase the application of a small baseline vision system for in-vivo microscale 3D reconstruction of capillary veins and develop a system for real-time analysis of microvascular blood flow for critical care. We believe that the computational methods developed in this thesis would find more applications of compact 3D sensing under challenging conditions.

Acknowledgments

I wholeheartedly thank my advisors, Srinivasa Narasimhan and Artur Dubrawski, for their guidance and support during my PhD journey at CMU. Their guidance inspires me through every single aspect of getting things work: from asking the right question to solving real problems; from fast idea prototyping to refining the implementation under real world conditions; from capturing the raw data to giving good presentations. The experience and lessons are valuable references that are sufficient for me to frequently revisit and get inspiration from in the future.

I'm also thankful to the rest of my thesis committee, Aswin Sankaranarayanan and Manmohan Chandraker, for making their time to be my thesis committee members. In addition, I thank Aswin for his helpful feedbacks and discussions regarding my thesis. I appreciate Manmohan for his willingness for sharing his insights during the CVPR Doctoral Consortium.

While working from an office at Smith Hall, I appreciate the chance of talking to Ioannis Gkioulekas, who sits in the same building and keeps his office door opened for insightful and helpful discussions. His insights, rigorous thinking, and passion always serve as a good example.

Parts of the thesis won't be possible without the collaborations with the exceptional medical doctors: Howard Edington, Micheal Pinsky and Hernando Gomez. The privilege of working with them granted me the chance of capturing real data in operation rooms, and validating the proposed pipelines on bedside data. More importantly, their expertise helps me to localize my thesis research into a gigantic context: developing new tool to help medical doctors save life.

I appreciate my mentors during the internship at Nvidia Research: Jan Kautz, Jinwei Gu, and Kihwan Kim, for spending a wonderful summer together. The time is fun also thanks to the connections with Ben, Soumyadip, Inchang, Matthias, and Chen.

One of the advantages of being co-advised is that I can work with the wonderful colleagues from two labs: the AutonLab and ILIM lab. Anthony, Jarod, and Predragp are always reliable when I want to turn the Matlab code in my laptop into the efficient compiled implementation on the server. Robert, Supreeth, Minh, Joe, Mark, Adithya, Tianchen, Shumian, Suren, Hiro, and Dinesh, thank you all for being great ILIM labates and helping with experiments. In addition, as part of the Computational Imaging group at CMU, I really enjoy the discussions and collaborations with Vishiwa, Jiayin, Rick, Yi and Jian.

Weichiu, Chen, Vincent, Phoebe, Si, Juyi, Haopeng & Man, Bob, Chendi, Bacon, thank you all for making the weekends fun. Thank you, Haopeng, for your kind help during my first months at Pittsburgh.

Last, but most importantly, I extend my gratitude to my family - my grandmother and parents. None of the experience or content in the thesis would be possible without your encouragements and unconditional support.

Contents

List of Figures

List of Tables

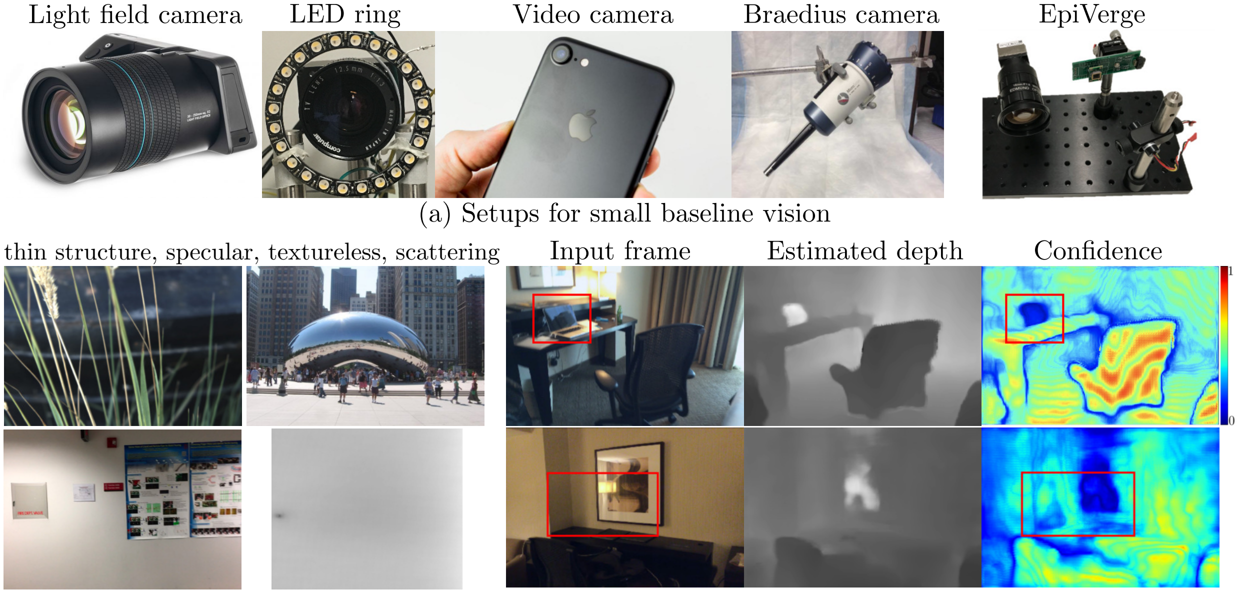
Chapter 1

Introduction

Portable camera sensor systems are becoming more and more popular in computer vision applications such as autonomous driving, virtual reality, robotics manipulation and surveillance, due to the decreasing expense and size of a RGB camera. We want such systems to be smaller so they can be deployed broadly in many applications. For example, the stereo cameras on the back of cellphone camera are about $\sim$1 centimeter apart; in the active illumination systems such as Kinect, the camera-to-projector distance is within $\sim$ 10 centimeters; in the light field cameras for VR, the micro-lens array is distributed within the diameter of a regular DSLR camera lens. The small baseline sensor structure is also common in the biological world. For example, the compound eye of Trilobite consists of multiple lenses with diameter up to 50 μm that are distributed on a 10 $mm \times 4$ mm region, and only a small number of such lenses could detect the potential predators [160]. For the aforementioned systems, the sensor baselines are usually within several centimeters or even millimeters, while the target working range is usually in meters.

Despite the compactness and portability of the small baseline vision systems, there are several limitations for such systems. First, it is well-known that the uncertainty in range finding using multiple views and the sensor baselines are inversely related. For small baseline vision systems, this means high depth uncertainties even for close range objects. Second, the cross-entropy between the images taken from different views is large. This makes the multiple measurements from a small baseline system redundant. This redundancy also reduces the robustness to the occlusion: if one region is occluded in one view, then it is very possible that the same region is also occluded in the other views. In other words, it is difficult to make inference, such as estimating the depth, for the occluded regions in one view.

On the other hand, besides compactness, a small baseline vision system has several advantages. First, compared to the regular baseline multi-view stereo system, it is easier to make correspondence due to the smaller search space and thus less ambiguities during finding the cross-view correspondence. This leads to simple and robust depth estimation with small baseline cameras, where it is sufficient to use the image intensities as the matched features for finding the per-pixel correspondences. Second, for multiple images captured across different views with small baselines, it is possible to use the cross-view differentiation based methods, such as optical flow, to infer the scene geometry and property [24, 200]. For multiple light sources with small baselines, it has been shown that we can estimate the material property, such as the Bidirectional Reflectance



(a) Setups for small baseline vision

(b) Scene complexity

(c) Estimation uncertainty

Figure 1.1: Three aspects of small baseline visions: (a) The small baseline distance can be either implemented as camera(s) with small baseline or light sources with short distances in between. From left to right: light field with micro-lens array; LED ring with small diameter; monocular video camera; Braedius camera [43]; EpiVerge [111] (b) Real world scenes are complicated due to phenomena such as occlusions, fine-grained structure, specular/textureless surface, and light scattering. We develop methods that take advantage of small baseline vision systems to handle those challenges. (c) For depth estimation, the depth uncertainty and the baselines are inversely related. As a result, in a small baseline setup, the depth estimation for far-away objects has very high uncertainties. The uncertainty may also come from other sources such as occlusion and specular reflection. We show that with small baseline camera setups, those uncertainties can be evaluated and reduced by sequentially accumulating frames captured from a monocular video camera.

Distribution Function (BRDF), by utilizing the change of the image intensities while changing the camera positions by a small amount [25]. Third, the methods using multi-view images, such as structure from motion, may benefit from the large overlapping regions among camera views with small baselines. For example, one can fuse the depth estimations from multiple images captured with small camera translations in order to improve the accuracy at the overlapping regions [135, 220].

1.1 Motivation

Given the aforementioned challenges and opportunities originated from small baseline vision, we ask: *How to use those advantages for small baseline vision setup while avoiding the limitations as much as possible?* Answering the question is important because it can not only help to deal with the aforementioned limitations for small baseline vision systems in order to broaden its applications, but also make some tasks that are difficult for large baseline systems easier. In this thesis, we approach this question in terms of three aspects as shown in Fig. 1.1:

Baseline distance in the setup The small baseline distance can be either implemented as camera(s) with small baseline or light sources with short distances in between. In each setup, we will take advantage of the benefits originated from the small distances between cameras/light sources for tasks that are much more difficult for large baseline setups.

Scene complexity Real world scenes are complicated due to phenomena such as occlusions, fine-grained structure, and light scattering. We explicitly formulate those effects in the forward image formation model. Then given the images, we solve for the inverse problem to get the estimations for surface shape or object depth. We have designed new methods to solve the inverse problems by taking advantage of small baseline setups.

Estimation uncertainty For depth estimation, the depth uncertainty and the baselines are inversely related. As a result, in a small baseline setup, the depth estimation for far-away objects has very high uncertainties. In addition, the uncertainty may also come from other sources such as occlusion and specular reflection. We show that with small baseline camera setups, those uncertainties can be estimated and reduced by sequentially accumulating multiple measurements.

1.2 Imaging Setups for 3D sensing

Depth sensing is crucial for 3D reconstruction and scene understanding. The depth sensing approaches can be categorized into passive and active ones, based on whether there is additional light source included in the imaging system and emits programmable light onto the scene. Here we take a brief discussion of the popular imaging methodologies with baselines designed for 3D sensing. We will consider single view depth estimation and imaging system without baseline such as Lidar as types with baseline equal to zero.

1.2.1 Passive Methods

For passive 3D sensing methods, the imaging system consists of one or multiple image sensors and optical components. Compared with active methods, the passive method is more energy efficient and robust to global illuminations such as light scattering. In addition, the prevalent availability of images and videos captured with passive imaging system enables learning-based depth estimation methods, where the model can either be trained with ground truth depths captured along with RGB images, or using photometric and geometric constraints when the ground truth depth is not available. Here, based on the camera setup and imaging scheme, we classify the passive methods into four categories: multi-view setup with known camera poses, multi-view setup with unknown camera poses, singe-view static setup with finite aperture capturing a stack of images with different focal settings, and singe camera capturing a single image.

Multi-view with camera poses When more than one images from different views of the scene are available, we can get the 3D structure of the scene using geometrical constraints such as triangulation. In the multi-view setup, the position and orientation of the cameras are known either from calibration or through a separate pose estimation process. The completeness and granularity of the recovered 3D structure increases with the number of views[163]. For a two-view (*i.e.* the stereo) setup, usually a depth map for one of the views is estimated [28]; for more than two view setups, a dense volume or mesh reconstruction is available [214, 220], providing more geometric structure information about the scene such as surface normal, which can be used for other tasks such as appearance capture and scene relighting.

In addition to pre-defined 3D representation such as point cloud, Signed Distance Function (SDF) and 3D volumes, recent methods [144, 171] use CNNs to learn the optimal representation to encode the 3D structure. The encoded representations have shown to be more compact and able to achieve higher resolution at the same time. Following this idea, it has been demonstrated that the view synthesis can be performed efficiently with a learned volumetric radiance field [127].

Multi-view without camera poses Given the image sequence captured by a video camera without pose measurements from motion sensors such as IMU, the 3D structure of the scene and the camera poses can be recovered with the scale ambiguity. Due to the global bundle adjustment where the pose and 3D structure are optimized simultaneously [53, 162], these methods are often computationally expensive. To reduce the computational expense, the noisy measurements for camera poses from inertial sensors such as CMU is incorporated into the SLAM system to provide better initial pose estimation hence faster optimization convergence. Because the IMU sensors are very compact and easily available at low price, the application of IMU is a practical way to improve the system efficiency, making the large-scale real-time applications possible [117].

As the pose estimation is based on either sparse feature correspondences among images [53, 184] or dense photometric difference among images [135], the performance degrades in the presence of textureless regions, occlusions and specular surfaces. An alternative approach to recover the camera pose from the image sequence is to formulate the pose estimation as a regression problem and learn the mapping from an RGB image to the 6D camera pose [18]. However, the generalization ability of such methods still remains to be validated.

Defocus For a camera with finite aperture, one pixel at the image plane receives radiance contributions from multiple rays from the 3D points within a double-sided cone determined by the focal plane and aperture size. When the imaged object is close to the camera with a finite aperture, the defocus effect comes into effect depending on the distance between the 3D points and the focal plane. The Point Spread Functions (PSFs) for the 3D points close to the focal plane approaches the delta function with diameters less than one pixel width, hence the image regions for those points are clear and have sharp edges. For 3D points that are outside the depth of field, their PSFs have non-zero support with an area larger than one pixel, hence the image regions are blurred. The distance between the focal plane and the camera can be controlled by adjusting the aperture size and focal length. By capturing a stack of images with different focal settings, and assuming the scene is static during the process, we can estimate the scene depth by examining the sharpness [202, 203] or degree of blur [77, 78] of the image region across different focal settings. For dynamic scenes, the same depth estimation scheme can be applied by using cameras with high speed sweeping focal planes [126].

Apart from using a conventional finite aperture lens, coded aperture has been used to generate PSFs with non-zero support in high frequency domains in the Fourier space [104]. This leads to better deblurring results since high frequency information (such as edges) are retained in the blurred images captured with the coded aperture. By taking one single blurred image with the coded aperture, it is shown that we can simultaneously recover the deblurred sharp image and the depth at the same time.

Single image With the large amount of RGBD images either collected by synthesizing [62, 76] or using dedicated depth sensors [41, 64, 131, 168], it is possible to learn the mapping from an RGB image to a dense depth map. In those methods, a black-box model (*e.g.*CNN) is trained either supervised by RGBD images [60, 67, 221] or unsupervised using photometric consistency constraint [221] and/or geometrical constraints [199].

1.2.2 Active Methods

For active depth sensing methods, the device consists of both light sources and sensors. Light is emitted from source towards the observed scene and the sensor captures the reflected light from the scene. Compared with the passive methods, the active methods perform much better on surface lack of textures by projecting coded illumination onto the scene. In addition, by configuring the distribution of energy for the light source, the sensing range in active methods is much larger than the passive ones. For example, by concentrating the power into one beam and using short-wave infrared (SWIR) wavelength, the Lidar system can increase the sensing range to over 200 meters [1]. On the other hand, due to the usage of light source, its performance degrades in the presence of global illumination (e.g. inter-reflection and scattered light), light source interference, ambient light and scene motion. Various approaches have been proposed to ameliorate those effects.

Based on the how light source is mounted and controlled, we classify the active methods into three types: triangulation-based (projector-camera stereo pair such as Kinect v1, Realsense) where the light source is fixed, angular-based (e.g. photometric stereo) with the light source position varies during data capture, and methods based on light travel time (e.g. ToF and LiDAR sensors) with high temporal resolution sensor measuring the travel time of the emitted light.

Triangulation The active triangulation methods use the same principle in the passive stereo vision: after finding out the dense correspondences between two views with known relative pose, the depth for each pixel on either of the two views can be estimated by triangulating the rays from that pixel and its corresponding pixel in the other view. In active triangulation methods, the two views are implemented as a projector-camera pair, where the projector projects a set of coded patterns onto the scene. The camera captures the scene under coded illuminations. By decoding the pattern on the camera side, we can make dense correspondence between the camera and projector and perform triangulation to get the dense depth maps for both views.

With the ambient light such as sunlight, the coded illumination from the projector tends to be overwhelmed and cannot be discerned from the camera view. In addition, due to global illuminations such as inter-reflection of concave objects, light scatterings due to fog or smoke, the illuminated patterns are severely blurred, which can also degrade the correspondence and triangulation performance One way to do alleviate this issue is to separate the direct and global illumination components [132] and then only the direct component for correspondence. Another more efficient way is to rule out as much global component as possible during capturing the image. This can be done by configuring the imaging scheme with a pair of synchronized rolling shutter camera and raster scan laser projector [142]. During the synchronized scan for the projector and camera, the projector scans through its epipolar line on the projector plane, at the same time, the camera opens its exposure for the corresponding epipolar line on the image plane. In this imaging scheme, only light coming from the same epipolar plane determined by the scan is captured and all the global illumination outside the plane is eliminated during capture. As a result, the system is robust to global illumination and strong ambient light.

Angular The 3D shape of an object can be recovered using its shading appearance. The photometric stereo methods recover the surface normal using images of the object captured with light sources from different directions. If there is no ambient light, by illuminating the object with distant light sources from different orientations and assuming that the object surface is Lambertian, the intensities of the reflected light can be modeled by the dot product between the surface normal and the light source direction. In this simplest case, the surface normals can be easily recovered by solving a linear system if the light source directions are known. However in most cases, the assumptions such as Lambertian surface and distant light sources does not hold true. For example, for a point light source (e.g. LED) close to the object, the light intensity falls off in an inverse squared manner. In addition, for non-Lambertian surfaces such as metal, the reflected light intensity changes with both the incident light direction and viewing direction. The light source positions may also be unknown. Various photometric stereo methods have been proposed to handle the non-Lambertian surfaces [70], point light source [151], and unknown light source [9]. The mapping between the photometric images and surface normals can be learned from synthetic images and applied on real images [155].

The Diffuse Optical Tomography (DOT) is a notable set of methods where the subsurface 3D structure embedded within a highly scattering medium (such as human skin) is recovered. In DOT, a grid of source-detector pairs are mounted on the tissue surface. Due to the embedded objects such as blood veins and tumor, the absorption coefficients vary across the tissue volume. In DOT, the 3D distribution of the absorption variation (hence the 3D structure of the embeddings) is recovered from the source-detector pairs mounted at different positions on the surface [167, 169, 219], assuming the scattering coefficients of the embeddings are close to the surrounding homogeneous tissues (e.g. human skin tissue).

Time-of-Flight Based on the method used for measuring the light travel time, the time-of-flight sensors can be categorized into two types: the ones directly measure the light travel time and the ones measure the phase shift of amplitude modulated signal.

To measure the light travel directly, an extremely short impulse (a few nanoseconds) is emitted from the light source. The sensor measures the time when the emitted photons are reflected back. As light travels at a speed of $3 \times 10^8 m/s$, the temporal resolution of the sensor (hence the clock) should be high enough to discriminate between extremely short light travel times in order to get a decent resolution in depth sensing. In addition, the sensor should be highly sensitive in order to detect the weak reflected light from far away objects. The above two requirements make the fabrication of the sensors that directly measure the light travel time complex and expensive. As a result, the ToF sensors of this kind (*e.g.*LiDAR) are usually with much lower resolution compared with the time modulated based sensors. However, due to the high sensitivity of the sensor (*e.g.*SPAD), the sensing range can be up to hundreds of meters.

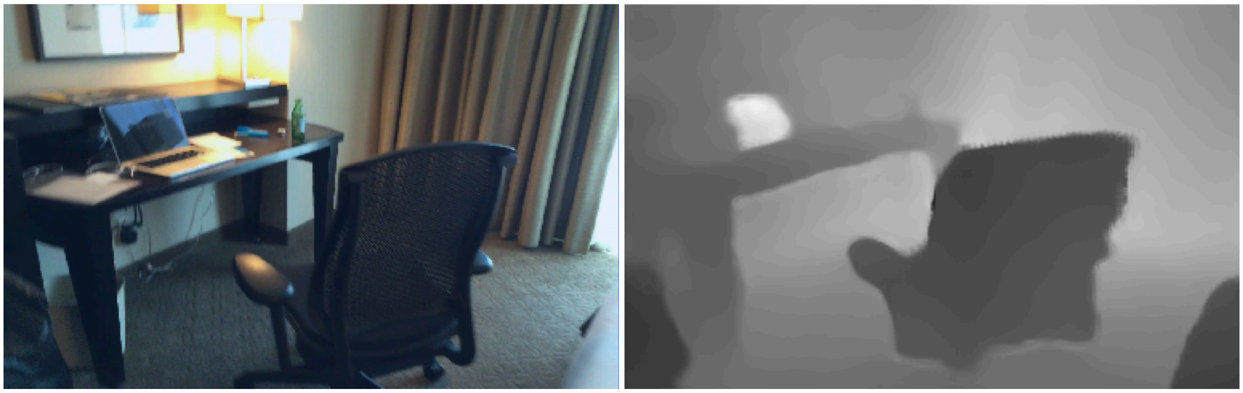
Rather than using a short impulse of light, the continuous wave time-of-flight (CW-ToF) systems generate amplitude modulated light. The scene depth can be estimated from the phase shift between the emitted light and reflected light. Since the measured phase shift is proportional to the modulo between the scene depth and the wavelength of the modulation, directly calculating the scene depth from the phase shifting results in depth ambiguity. This essentially wraps the distance into the sensors non-ambiguity range. To disambiguate the depth, the phase unwrap process is needed [47, 124]. As the fabrication of amplitude modulated source and phase sensors is easy, the CW-ToF systems are much cheaper, and more compact (*e.g.*Kinect v2 and ToF cameras on cellphones).

1.3 Challenges

The triangulation based and angular based 3D sensing system tends to be unreliable if the baselines between the light sources or the cameras become very small compared with the sensing range. For example, the distance of a video camera position between two adjacent frames is small if we want to use those two frames to estimate the scene depth. The space between two LEDs mounted on a small ring around the camera, as commonly seen in indoor surveillance cameras (*e.g.*Amazon Cloud Cam), is only a few centimeters. While the objects in the indoor environment are usually placed meters away. The scene complexities, such as occlusion, texture or specular surfaces, and scattering medium, also add difficulties for both active and passive systems. We will briefly discuss those challenges.

Small baseline For passive multi-view triangulation based methods, it is well known that the 3D estimation certainty is inversely related to the camera baselines. However, it is still to overcome the lack of camera baselines by using the rich redundancy in multiple images for large overlapping fields of views. For example, with the tiny accidental handshake during capturing a video, it has been shown that the dense depth map can be recovered [217] by utilizing the depth correlation among spatially close pixels with similar appearances. Global bundle adjustment has been applied on a small baseline video clip in [87] to find the correspondences among multiple frames captured with a rolling shutter camera. For 3D sensing of far away objects, a three-camera setup with small FoVs is designed [93]. With a camera baseline up to 2 meters, the reliable sensing range of the small FoV camera setup can be above 200 meters, surpassing the LiDAR sensors. An extreme case is where the camera is static and the imaging setup degrades as a single image depth estimation where the baseline is zero and the depths are estimated with other information such as ordering [60], appearance correlation among pixels [115], semantic information [109], object boundaries [83], and so on.

For active methods using multiple light sources such as photometric stereo, a small baseline among the light sources leads to very subtle variation of image intensities across images, making the photometric stereo more difficult. With zero baseline, the photometric stereo degrades into a shape from shading problems where only one single image is available. However, for small light direction variations, the derivative of image intensity w.r.t. the light position has been shown to be useful enough for surface reconstruction [35]. In addition, under distant illumination assumption, the differential image intensity is related to the surface geometry regardless of its isotropic BRDF. The relation can be used to recover the surface normal and depth for objects with unknown,



(a) Input frame

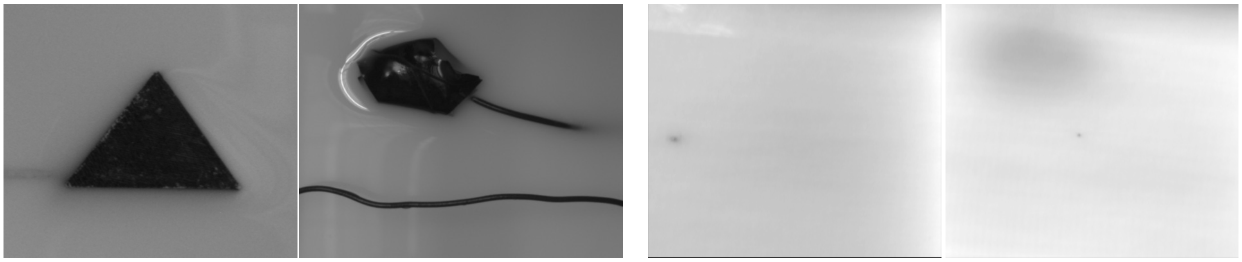
(b) Estimated depth

Figure 1.2: Specular surface poses a challenge for passive depth estimation, especially for methods using multi-view triangulation where the surfaces in the field of view are assumed to be Lambertian. (a) One of the input frames from a monocular video camera. The laptop monitor is specular. (b) The estimated depth map for the input frame. The depth values at the specular region is inaccurate due the failure of Lambertain assumption in triangulation.

isotropic BRDF reflection properties [25].

Occlusions With the presence of occlusions, traditional triangulation based methods, such as structure from motion and coded illumination, fail to work well on scenes due to lack of correspondence or block of active illuminations. For an image where the background is blocked, the occluders, such as rain drop or stain on the windows, can be removed using learning-based method [51], where the model for mapping the corrupted image to the occlusion-free image is trained with a set of pairs of images with and without occlusions. If multiple images of the occluded background image is available (*e.g.*captured with different focal settings or different camera positions), one can remove the occlusions using the redundancy among images [73, 211]. With active light, it has been shown that we can reconstruct the 3D structures of the occluders without correspondence matching by taking advantage of the fact that the occlusion blocks the illumination and cast shadows[212], or absorbs part of the illumination with absorption variation within the scattering medium [38].

Textureless/Specular Surface In triangulation-based passive 3D sensing method, the performance of correspondence matching depends on two major conditions: (1) the appearance feature is informative and discriminative so the feature matching across frames is reliable; (2) the change of appearance feature across frames is limited (*e.g.*view-independent Lambertian surface). However, for textureless surfaces such as a white wall, the appearance feature is usually not discriminative due to lack of image intensity gradient; for specular surfaces, the appearance feature for the same 3D location in the scene changes across frames. As a result, the depth values estimated for the textureless or specular surfaces in the scene are error-prone, as shown in Figure 1.2. For passive methods, one way to deal with a textureless surface is to use a larger receptive field, as in the current state-of-art CNN-based methods, in the hope that it includes the adjacent textured regions (*e.g.*textured floor below a white wall). Then fill the depth values in the textureless region



(a) medium free (b) whole milk

Figure 1.3: Due to dense subsurface scattering, the heterogeneities underneath the whole milk surface are hard to observe or invisible under daylight. (a) The medium free images where the heterogeneities are positioned above the surface; (b) The heterogeneities are beneath the whole milk and can be hardly discerned.

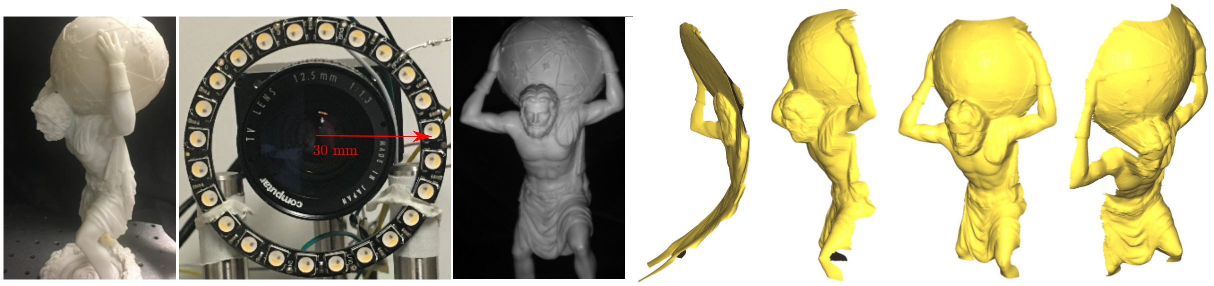
.

using spatial correlation using CRFs [31]. To handle the specularities, new feature correspondence metrics, rather than photometric consistency, has been proposed for images captured with densely angular-sampled views (*e.g.*using a light field camera), while taking into account the non-Lambertian surface properties [180]. An alternative approach is to place a polarizer in front of the sensor to remove the specularities during capturing the images [172].

Subsurface Light Scatterings For the scatter medium, part of the incident light reaches into the surface, scatters randomly before leaving the medium and being captured by the image sensor. During each scatter event, the light radiance changes due to absorption, scattered light into/outwards the propagation direction. As a result, in addition to the surface reflectance properties, the intensity of the scattered light also depends on the scattering properties within the medium. The light radiance in the scattering process can be modeled using the Radiative Transfer Equation (RTE) [27]. However, the RTE is recursive and there is no analytical solution for the light radiance term. For physically accurate modeling of light radiance in scattering, the computationally expensive Monte Carlo method is implemented to solve the full RTE. In graphics, solving the RTE efficiently for fast rendering of scattering medium requires certain simplification for the scatter medium, such as layer representation for human skin [45], isotropic scattering phase function for material and point light source for illumination [90], or directional ray [59]. In this case, the RTE is approximated with models where the solution is analytically available and efficient to compute.

In the inverse problems where we want to extract the scene information ,additional simplification has been made. For example, for fog and smoke, it is assumed that the single scattering event dominates. In this case, the attenuation of light radiance due to scattering can be analytically formulated and the optical distance and scene depth can be estimated from images captured in foggy environments [130]. For image dehazing application [80], even more simplification is made such that only light attenuation due to scattering is considered.

For dense medium where the single scatter model fails, the isotropic scattering phase function simplification has also been made in order to solve RTE analytically in DOT systems [223]. Recently, it is shown that the RTE can be fully taken into consideration by differentiable Monte Carlo rendering [218]. The position of the inhomogeneous embeddings within a highly scatter medium



(a) Object (b) Light sources (c) Input image (d) [151] (e) Our method

Figure 1.4: Near-Light Photometric Stereo using LED light sources placed in a planar circular ring centered around the camera lens (a) The profile for the reconstructed object; (b) Our imaging setup with a 30 mm radius ring of 24 LEDs; (c) One of the 24 input images; (d) Due to non-convexity of the near-light photometric stereo problem, reconstruction using [151] fails for depth initialization far away from the true values. (d) Reconstruction using our two-stage method that directly optimizes a 3D mesh.

(*e.g.*tumor underneath skin) can be estimated given the input image, by differentiating the unknown position w.r.t.the modeling error. However, due to the high computational load and non-convexity involved in solving the full RTE, the data dimension of the unknowns is limited. In this thesis we propose an imaging system that captures high resolution subsurface images, as shown in Figure 1.3, and an efficient tomography method to recover the 3D structures embedded in dense medium. Our method is computationally efficient and convex. So it can be used as the initial bootstrap for methods where the full RTE is considered [218].

1.4 Goals and Contributions

In this thesis, we aim to design computational hardwares and algorithms where the small baseline setup is utilized to deal with the scene complexities in order to apply 3D sensing in challenging environments in the real world, as listed in Tab. 1.1. Towards this goal, we make the following contributions:

Small Baseline Photometric Stereo [114]: In Chapter 2, we utilize the small light source baselines. We propose a two-stage near-light photometric stereo method using circularly placed point light sources (commonly seen in recent consumer imaging devices like NESTcam, Amazon Cloudcam, etc), as shown in Fig. 1.4. In the first stage, we optimize the vertex positions using the differential images induced by small changes in light source position. This procedure yields a strong initial guess for the second stage that refines the estimations using the raw captured images. We also propose an accurate calibration approach to estimate the positions of the sources.

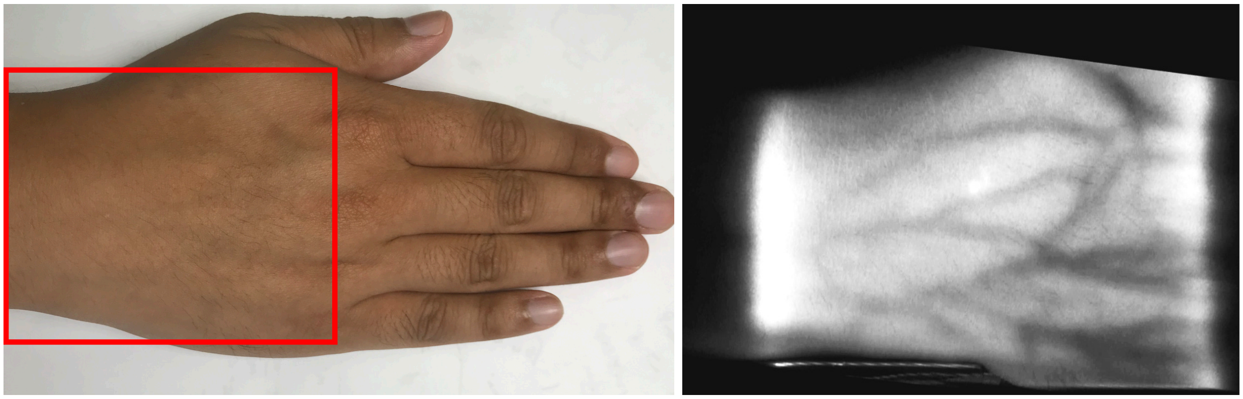
Multilayer Thin Structure Reconstruction [113]: In Chapter 3, we take advantage of the small camera baselines that makes the correspondence easier. We present a method for matting and depth recovery of 3D thin structures with self-occlusions using a single-view camera with finite aperture lens. To this end, we propose an image formation model that explicitly describes

	Small Baseline Photometric Stereo	Multilayer Thin Structure Reconstruction	Monocular Depth Estimation with Uncertainties	EpiVerge
Setup				
Passive/Active	A	P	P	A
Range to Baseline Ratio	∼100	∼10	∼50	∼500
Scene Complexity				
Opaque Occlusion	✗	✓	–	✓
Specular	✗	✓	–	–
Textureless	✓	✗	–	✓
Medium Scatter	–	✗	✗	✓
Uncertainty	✗	–	✓	–

Table 1.1: Comparison of methods presented in this thesis based on the imaging setup, scene complexities handling, and whether the uncertainty estimation is available. The ratio numbers are roughly calculated based on the typical camera/light source baselines and the sensing range. For EpiVerge, the minimal line separation is around 0.03 mm, while the volume we are reconstructing spans around 15 mm in the depth axis. The notations - ✓ : The scene complexity is handled either by specifically modeled or using hardware design that is robust to it. For uncertainty estimation, it denotes it is directly available. – : The scene complexity is partially handled or does not impact negatively on the performance, and uncertainty can be reflected in the output, although not directly available. ✗ : The method is not robust to the scene complexity, and the output does not include any information about the estimation uncertainty.

the spatially varying optical blur and mutual occlusions for structures located at different depths. Based on the model, we derive an efficient MCMC inference algorithm that enables direct and analytical computations of the iterative update for the model/images without re-rendering images in the sampling process.

Monocular Depth Estimation with Uncertainties [110]: In Chapter 4, we focus on videos captured by a monocular RGB camera in motion, in which case the camera baselines for adjacent frames are small. More specifically, we propose a deep learning (DL) method to estimate per-pixel depth and its uncertainty continuously from a monocular video stream, with the goal of effectively turning an RGB camera into an RGB-D camera. Unlike prior DL-based methods, we estimate a depth probability distribution for each pixel rather than a single depth value, leading to an estimate of a 3D depth probability volume for each input frame. These depth probability volumes are accumulated over time under a Bayesian filtering framework as more incoming frames are processed sequentially, which effectively reduces depth uncertainty and improves accuracy,



(a) Skin under daylight (b) Short-range indirect image

Figure 1.5: (a) Due to dense subsurface scattering, the blood veins underneath the skin is hard to observe or invisible under daylight. (b) The find grained structure of the subsurface blood veins are clearly visible under the short-range indirect image captured by EpiVerge [111]. The subsurface skin image is useful for medical application such as skin disease diagnosis, biopsy targeting, and needle injection.

.

robustness, and temporal stability.

EpiVerge [112]: In Chapter 5, we focus on reconstructing the 3D structure of the heterogeneous inclusion within highly scatter medium such as whole milk and human skin. The imaging system consists of a high-resolution camera and a laser projector. This proposed imaging system enables us to see through densely scattering medium and observe fine-grained structures such as veins underneath the surface, as shown in Fig. 1.5. On top of the system, we present an efficient algorithm for high resolution diffuse optical tomography with a scanning line imaging and illumination pair setup.

Real-Time Capillary Vein Blood Flow Analysis [29]: In Chapter 6, we showcase the application of small baseline vision system for real-time analysis microvascular blood flow for critical care. In this application the Sidestream Dark Field (SDF) imaging device has been used to visualize and support interpretation of the microvascular blood flow.

Chapter 2

Near Light Photometric Stereo with Small Baseline LEDs

2.1 Introduction

Recovering surface shape is important for a wide range of applications such as robot manipulation, cultural heritage digitization and skin surface analysis *etc*. Photometric stereo methods use shading cues from images captured with varying illumination to recover surface shape. Traditional photometric stereo methods assume that light sources are distant, thus the lighting directions for all scene points are parallel. This is true for light sources such as the sun or for indoor lights placed far away from small objects. Under the distant light assumption, we are able to linearly solve the surface normal given the image intensities with calibrated or uncalibrated light source directions.

However, the distant light source assumption fails when the object-to-light distance becomes small. In the near-light setting, the image intensity depends non-linearly on the 3D location and normal of the scene point as well as the 3D light source position. Furthermore, for objects close to a perspective camera, the widely assumed orthographic projection model also fails. In this case, the relation between the surface normal and the depth of a scene point, which are often defined in image coordinates (at each pixel), becomes more complex when back-projected to 3D. Thus, solving for the 3D shape of an object that is illuminated by near light sources and that is captured by a projective camera is a highly non-linear and non-convex problem. As a result, photometric reconstruction often fails without strong initial guesses. , as shown in Fig.1.4 (d).

In this chapter, we present a near-light photometric stereo algorithm with circularly-placed point light sources and a perspective camera. This algorithm includes three novel contributions. First, we model the scene as a 3D triangulated mesh whose vertices correspond to the observed pixels, and directly optimize the positions of the vertices. The key advantage of this representation is that the vertex normals can be simply computed using adjacent triangular faces of the 3D mesh. The alternative of representing surface normals as numerical derivatives of depths in image coordinates (e.g. $N = (z_x, z_y, 1)$) results in unnecessary complexity when back-projected to 3D. Second, we split the algorithm into a two-stage process. In the first stage, we solve photometric stereo using the differential images captured by changing the light source position in a small amount along a circular path. We show that the analytical form of how the vertex position is related to measured differential intensity is less complex and results in reliable estimates in most parts of the object. In the second stage, these vertex positions are refined using the original image formation model applied to the raw captured images.

The above algorithm is still sensitive to errors in calibration. The light source positions in 3D are often obtained using multiple specular spheres of known radii and locations[106, 148]. But the 3D positions of these specular spheres are hard to measure accurately, resulting in poor localization of the sources for the proposed algorithm. Thus, as a third contribution, we present a simple calibration approach that uses a flat panel display to estimate the source positions. The flat panel display specularly reflects light, and unlike spheres, can be calibrated precisely for position and orientation using camera calibration methods [92].

Together, the three contributions lead to effective performance on both synthetic and real scenes with complex shapes placed at various distances from the source/camera. Our method outperforms previous state-of-the-art in near-lighting photometric stereo, where the optimization suffers from poor initial guess. Our approach also outperforms distant-light photometric stereo methods, even when the distance of the object is several times (5X-10X) than the radius of the LED ring. As a side effect of using differential images, our method tends to perform better in the presence of diffuse inter-reflections (but we make no claim on eliminating these effects). Our system is portable and can be implemented using a small off-the-shelf LED ring. Thus, we believe this work is timely enabling photometric 3D reconstruction on consumer imaging devices like Cloud-cam, Nest-cam that increasingly use small LED rings for imaging nearby indoor and outdoor scenes.

2.2 Related Work

Photometric Stereo with Distant Light Sources: Since the first formulation of the Photometric Stereo problem in [205] for shape reconstruction, there have been numerous works on improving and generalizing the method by taking into account different aspects during image formation, camera calibration and light source variations. In [133], the shape is recovered using inter-reflections by modeling the inter-reflections with form factor. For translucent objects, subsurface scattering has been taken into account in [44] and [88]. The volumetric scattering for the under-water imaging scenario is modeled in [129]. A good survey and benchmark dataset can be found in [166]. Light intensity calibration error has been considered in [33]. The solution space for Photometric Stereo and the ambiguity in the recovered shape have been discussed in [11, 12].

Photometric Stereo with Near-field Sources: The parallel illumination direction assumption fails when the light source is close to the object. In this case, the light source is modeled as a point light source (quadratic fall-off) and the illumination direction depends on the 3D location of the scene point. In [151], a variational method is proposed to solve the inverse problem. In [143] and [107], the near-light photometric stereo is solved without calibrating the light source. In [201], a thorough analysis for reconstruction error in the near-light setup is performed. All these approaches are highly sensitive to initial guesses and do not use differential lighting based approach proposed in this work. In [206] and [187], the near-light photometric constraint is added to the multi-view scene reconstruction pipeline from images captured with different camera views.

In contrast, our work is based on a single perspective view.

Photometric Stereo using Differential Lighting: In [65, 72, 91, 118], gradient illumination implemented either with a light dome or a ring of LEDs is used for surface reconstruction of human faces. In [222], the ring LED setup is used as an additional constraint during reconstruction. In [26], the differential motion of light source in the 1D circular trajectory is used for reconstructing surface with unknown BRDF. But in all these works, the sources are assumed to be distant. Most closely related is the work of [35] where the scene depth is solved directly using images captured with small near-field point light source motion. However, in order to solve for the scene depth, 3 motion directions are needed for each light source position. In contrast, the trajectory of the point light sources in our case is just a 1D curve, *i.e.* a planar circular ring, with 2 degrees of freedom less for the light source motion compared to [35].

Our method is closely related to Xie *et al.* [208], where the near-light photometric stereo problem is formulated in terms of mesh deformations. Our method is different from [208] in three aspects: (1) We model the scene with 3D triangular meshes and optimize the depths of the vertices directly. The surface normals are determined directly from the 3D positions of the vertices. So there is only one variable for each pixel. Xie *et al.* [208] represent the scene as a rectangular mesh with both the surface normals and depths as variables (similar to many other previous works). So there are three variables for each pixel, making it harder to optimize. This is redundant since for a mesh representation, normals are completely determined by vertice positions; (2) The method in [208] assumes orthographic camera model. Thanks to the scene representation, our method works for perspective cameras; (3) Because we determine the surface normal from the vertice depths, our method does not rely on mesh deformation to get the depths from surface normals, as in [208]. This leads to robustness to depth discontinuities for our method.

2.3 Near-Light Photometric Stereo on a 3D Mesh

In this section, we describe the image formation model for near-light photometric stereo of a Lambertian object illuminated by point light sources and captured by a perspective camera. Without loss of generality, we set the origin for the world coordinate frame to be the center of the camera, as shown in Fig. 2.1. The albedo and the surface normal for a scene point are denoted by ρ and $\mathbf{n}$. The point source is at location $\mathbf{s}$. Then radiance R of a scene point at $\mathbf{x}$ is:

$$R = \rho s_e \frac{\mathbf{n}^T(\mathbf{s}-\mathbf{x})}{|\mathbf{s}-\mathbf{x}|^3} = \tilde{\rho}\frac{\mathbf{n}^T(\mathbf{s}-\mathbf{x})}{|\mathbf{s}-\mathbf{x}|^3}, \tag{2.1}$$

where s_e is the light source intensity; $\tilde{\rho} = \rho s_e$ is the scaled albedo. The cubic in the denominator accounts for the normalization for the incident light vector and the quadratic fall-off of light intensity in the point light source model.

The scene point is imaged by a camera with intrinsic matrix K. We define the homogeneous image coordinate for the point $\mathbf{x}$ projected on the image plane to be $\mathbf{p}$. For a scene point with depth z, the image coordinate $\mathbf{p}$ and the world coordinate $\mathbf{x}$ are related by back-projection:

$$\mathbf{x} = K^{-1}\mathbf{p}z \tag{2.2}$$

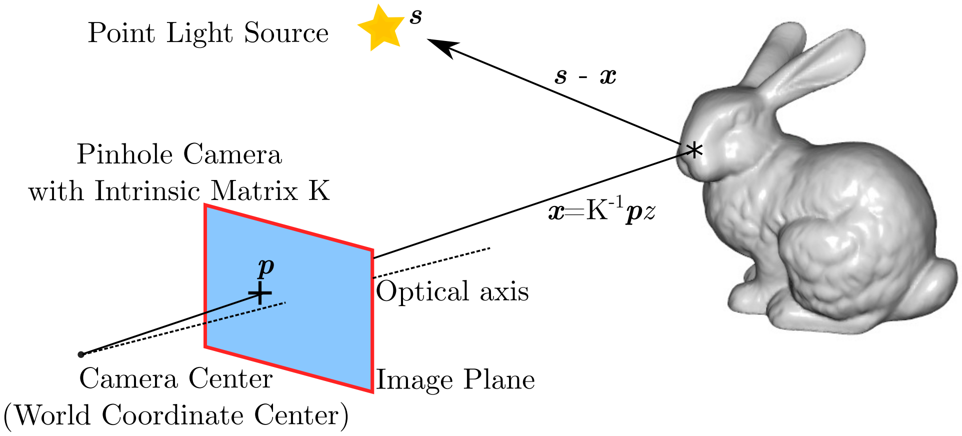


Figure 2.1: Image formation with near-field light source and a projective camera. For each pixel $\mathbf{p}$, there is only one variable depth $z(\mathbf{p})$. Given K, the position of the point in 3D is $\mathbf{x} = K^{-1}\mathbf{p}z$. Its surface normal $\mathbf{n}(\mathbf{p})$ is determined by $z(\mathbf{p})$ and the depths of its surrounding points as describe in Sec.2.3.

Combining Eq.2.1 and Eq.2.2, the image intensity $I(\mathbf{p}; z, \mathbf{n})$ for the scene point $\mathbf{x}$ can be written as:

$$I(\mathbf{p}; z, \mathbf{n}) = \tilde{\rho}\frac{\max\{\mathbf{n}^T(\mathbf{s} - K^{-1}\mathbf{p}z), 0\}}{|\mathbf{s} - K^{-1}\mathbf{p}z|^3}, \tag{2.3}$$

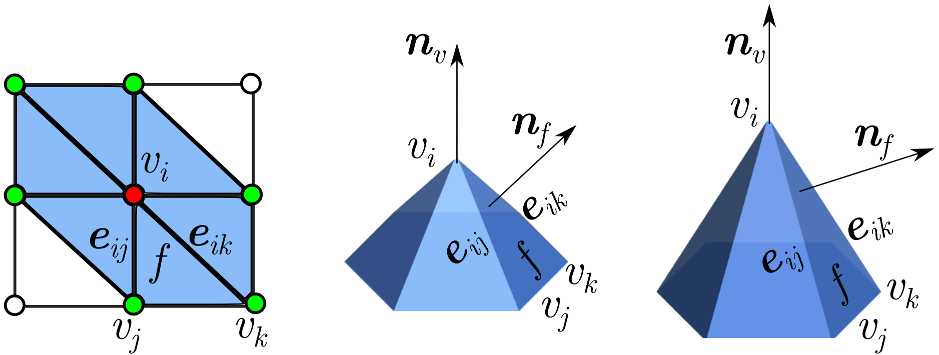
where, attached shadow is modeled using the $\max$ operator.

It is hard to optimize for surface normals and depths as separate unknowns. Thus, we need to exploit their relationship. However, representing surface normals as numerical derivatives of depths in image coordinates (e.g. $N = (z_x, z_y, 1)$) results in unnecessary complexity when back-projected to 3D. Instead, we represent the scene as a 3D mesh with triangular faces F whose vertices V are defined for all image pixels. We then use the vertex normal for calculating the image intensity in Eq. 2.3. This process is illustrated in Fig. 2.2(a). The 3D location of the vertex v_i is $\mathbf{x}_i$ and its 2D imaged location is $\mathbf{p}(v_i)$. Given the depth $z(v_i)$ for vertex v_i, $\mathbf{x}(v_i)$ is given by $\mathbf{x}(v_i) = K^{-1}\mathbf{p}(v_i)z(v_i)$. An adjacent face f consists of v_i and two other vertices v_j and v_k. The edges connecting v_i to v_j and v_k are $\mathbf{e}_{ij} = \mathbf{x}_j - \mathbf{x}_i$ and $\mathbf{e}_{ik} = \mathbf{x}_k - \mathbf{x}_i$ respectively. Then, the unnormalized vertex normal for v_i is defined as:

$$\hat{\mathbf{n}}(v_i) = \frac{\sum\limits_{f \in \mathcal{N}_f(i)} a(f)\mathbf{n}(f)}{\sum\limits_{f \in \mathcal{N}_f(i)} a(f)} = \frac{\sum\limits_{f \in \mathcal{N}_f(i)} [\mathbf{e}_{ij}]_\times \mathbf{e}_{ik}}{\sum\limits_{f \in \mathcal{N}_f(i)} a(f)} \tag{2.4}$$

where $\mathcal{N}_f(i)$ are the neighboring faces that include vertex v_i; $\mathbf{n}(f)$ and $a(f)$ are the normal and area for face f. The vertex normal $\mathbf{n}(v_i)$ is obtained by normalizing $\hat{\mathbf{n}}(v_i)$:

$$\mathbf{n}(v_i) = \hat{\mathbf{n}}(v_i)/|\hat{\mathbf{n}}(v_i)| \tag{2.5}$$



(a)Ring-1 neigbor (b)Face and Vetext normals (c)Ambiguity $\boldsymbol{n}(v)$

Figure 2.2: Local geometry around vertex v_i. (a) The surface normal at vertex v_i, colored in red, depends on the depths of its ring-1 neighborhoods, colored in green; (b) We define the surface normals to be the vertex normal $\mathbf{n}(v_i)$, which depends on the face normals of meshes sharing v_i (c) The original vertex normal definition leads to ambiguity in the depth estimation for v_i. We deal with this ambiguity using the method presented in Sec.2.3.

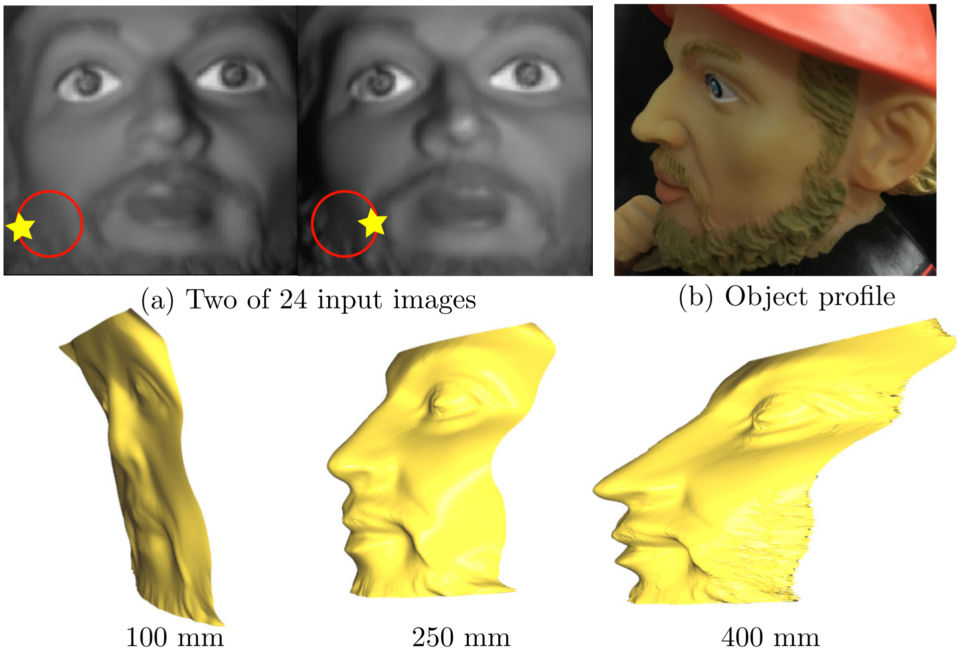
We solve for the depths of the vertices by combining image formation model in Eq.2.3, Eq.2.4 and Eq.2.5. However, there remains a depth ambiguity for vertex v_i if we use the normal definition in Eq.2.4. Consider the cases shown in Fig. 2.2(b) and (c). The faces around vertex v_i are related by rotations around the vertex normal $\mathbf{n}_v$. Because the face areas are the same, the horizontal components of the surface normals of the neighboring faces are canceled out in the weighted sum in Eq.2.4. So the vertex normals in two cases both point perpendicularly upwards, even though the locations for vertex v_i and the face normals are different.

We solve this ambiguity by changing the order of shading in Eq.2.3 and computing the weighted averaged in Eq.2.4: we first compute the image intensities for each face around vertex v_i, then compute weighted average of intensities to get the intensity for vertex v_i. More formally, the image intensity $I(v_i; z(v_i))$ for vertex v_i is re-written as:

$$I(v_i; z(v_i)) = \tilde{\rho} \frac{\sum_{f \in \mathcal{N}_f(i)} a(f) \mathcal{S}\left(\mathbf{s}, \hat{K}, z(v_i), f\right)}{|\mathbf{s} - \hat{K}(v_i) z(v_i)|^3 \sum_{f \in \mathcal{N}_f(i)} a(f)}, \tag{2.6}$$

with $\hat{K} = K^{-1}\mathbf{p}(v_i)$ and the shading operator defined as:

$$\mathcal{S}\left(\mathbf{s}, \hat{K}, z(v_i), f\right) = \max\{\mathbf{n}^T(f)(\mathbf{s} - \hat{K} z(v_i)), 0\}$$



(a) Two of 24 input images

(b) Object profile

(c) Surface reconstructions with different depth initilizations

Figure 2.3: Sensitivity of the optimization of depths of vertices to initial guesses with planes in different depths. The light sources are a 30 mm radius ring of 24 LEDs centered around the camera. The object is placed 200 mm away from the camera. (a) Two out of 24 input images, with the LED positions at the lower left corner. (b) The profile of the reconstructed area for reference. (d) The profiles of the surfaces reconstructed with different initial depths.

2.3.1 Objective Function

We estimate the depth values for all vertices by minimizing the difference between the modeled intensities I in Eq.2.6 and the measured image intensities $\tilde{I}$:

$$\begin{aligned} \min_{z}. \quad & \sum_{v_i \in V} \left(\tilde{I}(v_i) - I\left(v_i; z(v_i)\right) \right)^2 + \lambda_I E_s(z(v_i)) \\ \text{with} \quad & E_s\left(z(v_i)\right) = \sum_{v_j \in \mathcal{N}_v(i)} \left(z(v_i) - z(v_j)\right)^2 \end{aligned} \tag{2.7}$$

where $\mathcal{N}_v(i)$ is the set of vertices in the Ring-1 neighborhood of v_i. The albedo $\tilde{\rho}$ is solved analytically using Eq.2.6.

The energy function for the optimization problem defined in Eq.2.7 has a numerous local minima due to the cubic term in the denominator in $I(v_i; z(v_i))$ defined in Eq.2.6. So, good initializations of the depth values at vertices is crucial, as validated using experiment shown in Fig.2.3. The face of a toy is reconstructed using the L-BFGS minimizer for Eq. 2.7 with different initial depth values. We use a ring of 24 LEDs centered around the camera. The radius of the LED ring is 30 mm and the object is placed around 200 mm away from the camera. More details about the implementation are included in Sec.2.6. As shown in Fig.2.3, if the initialization of depths is far away from the true values, we get either over-flattened or stretched results due to the local minima.

2.4 Near-Light Photometric Stereo with Differential Circular Source Motion

In order to get a good initial guess for the optimization problem in Eq.2.7, we put forward to use the differential images induced by small change of light source position. The illumination and imaging geometry is shown in Fig. 2.4. For notation simplicity, for LED at position s and with index t, we denote the differential source motion as s_t and the corresponding differential image intensity as I_t, which means the differential values *w.r.t* to the LED index. By differentiating the image formation model in Eq.2.1, we get the analytical form for the differential image intensity I_t:

$$\begin{aligned} I_t &= \frac{\partial I}{\partial \mathbf{s}} \mathrm{s}_t \\ &= \tilde{\rho} \frac{\mathbf{n}^T \mathbf{s}_t}{|\mathbf{s} - \mathbf{x}|^3} - 3\mathbf{n}^T(\mathbf{s} - \mathbf{x}) \frac{\tilde{\rho}(\mathbf{s} - \mathbf{x})^T \mathbf{s}_t}{|\mathbf{s} - \mathbf{x}|^5} \\ &= \tilde{\rho} \frac{\mathbf{n}^T \mathbf{s}_t}{|\mathbf{s} - \mathbf{x}|^3} - 3I \frac{(\mathrm{s} - \mathrm{x})^T \mathrm{s}_t}{|\mathrm{s} - \mathrm{x}|^2} \end{aligned} \tag{2.8}$$

which can be simplified by writing the first term as a function of the image intensity I:

$$I_t = I \frac{\mathbf{n}^T \mathbf{s}_t}{\mathbf{n}^T(\mathbf{s} - \mathbf{x})} - 3I \frac{(\mathrm{s} - \mathrm{x})^T \mathrm{s}_t}{|\mathrm{s} - \mathrm{x}|^2} \tag{2.9}$$

Intuitively, the first term is the contribution of the change of source direction; the second term is due to the change of distance between the light source and the scene point.

For light sources mounted on a plane parallel to the image plane, there are two special cases where the second term including the inverse squared distance in Eq.2.9 becomes small and can be ignored.

The first case occurs when the angle between the light direction $\mathbf{s} - \mathbf{x}$ and the light source motion vector s_t is large such that $(\mathrm{s} - \mathbf{x})^T \mathrm{s}_t \approx 0$. This happens when the object is placed far away from the light source such that $\mathrm{s} - \mathbf{x}$ is perpendicular to the light source plane spanned by s_t. However, for large distance between the light source and scene point, both the captured image intensity I and the differential image intensity I_t would be too small with low SNR.

The second case where we can ignore the second term in Eq.2.9 is when the source motion trajectory is circular. Here, the source motion direction s_t and position s are perpendicular, thus $\mathrm{s}^T \mathrm{s}_t = 0$. So, the differential image intensity I_t in Eq.2.9 becomes:

$$I_t = I \frac{\mathbf{n}^T \mathbf{s}_t}{\mathbf{n}^T(\mathbf{s} - \mathbf{x})} + 3I \frac{\mathbf{x}^T \mathbf{s}_t}{|\mathrm{s} - \mathrm{x}|^2} \tag{2.10}$$

For a camera with a traditional field of view, the angle between the direction of line of sight and the normal of image plane is small. Since the plane spanned by s_t is parallel to the image

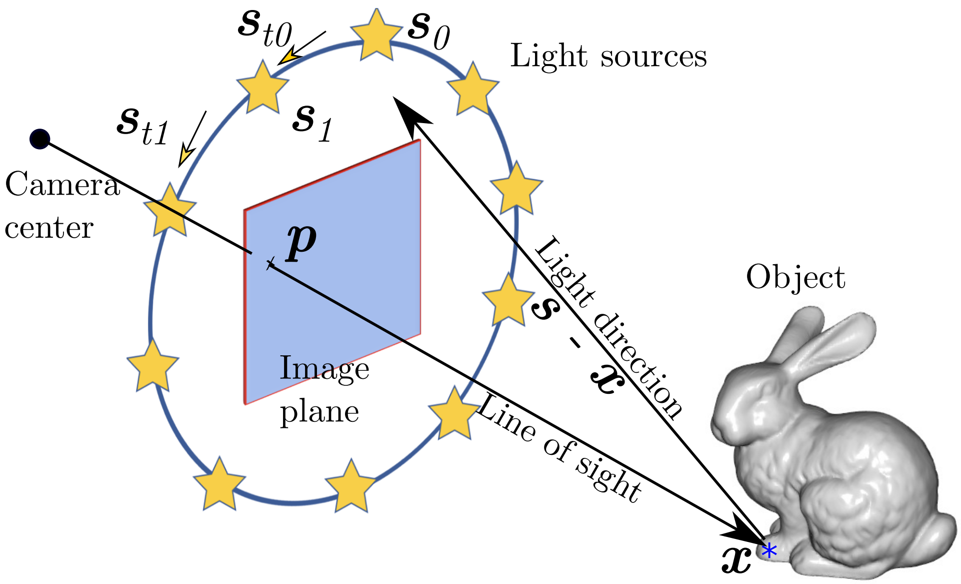


Figure 2.4: The geometry of differential change of light source positions. The light sources are densely mounted on a planar circle centered around the camera.

plane, $\mathbf{x}^T\mathbf{s}_t$ in the second term in Eq.2.10 becomes small. The term $\mathbf{x}^T\mathbf{s}_t$ is further attenuated by the squared distance between the point and the light source $|\mathbf{s}-\mathbf{x}|^2$. So we can assume that the second term in Eq.2.10 for small light source motions can be ignored, at least for the purposes of estimating our initial guess.

Thus, the differential image intensity and the measured image intensity are related by:

$$I\mathbf{n}^T\mathbf{s}_t - I_t\mathbf{n}^T(\mathbf{s}-\mathbf{x}) = 0 \tag{2.11}$$

This is similar to the differential image term in [26] for the Lambertian case. The difference is that we can solve for the depths using Eq.2.11 alone rather than obtaining a constraint for the surface normal as in [26].

More specifically, given the measured image $\tilde{I}$ and differential image $\tilde{I}_t$, the depths can be estimated by:

$$\min_{z}. \quad \sum_{v_i\in V} E_{It}(v_i;z(v_i)) + \lambda_{I_t}E_s(z(v_i)) \tag{2.12}$$

with

$$\begin{aligned} E_{It}(v_i;z(v_i)) &= \left(\hat{\mathbf{n}}(v_i)^T\left(\tilde{I}(v_i)\mathbf{s}_t - \tilde{I}_t(v_i)(\mathbf{s}-\mathbf{x}(v_i))\right)\right)^2 \\ \mathbf{x}(v_i) &= K^{-1}\mathbf{p}(v_i)z(v_i) \end{aligned}$$

Note that the energy function in Eq.2.12 is independent from the inverse squared distance. Thus the energy function is less non-linear than the one in Eq.2.7. In addition, since the function is independent from the albedo, given the measured image intensities $\tilde{I}$ and differential image $\tilde{I}_t$, we can estimate the depths without knowing the surface albedo.

2.5 Complete Algorithm and Calibration

We use the optimized depth in Eq.2.12 as the initial values in Eq.2.7 where we estimate the depth values with raw image intensity I. To initialize the optimization problem in Eq.2.12, we use a line search for the depth of each vertex. Given the candidate depth value $z_c(v_i)$, we can solve for the corresponding candidate surface normal $\mathbf{n}_c$ using Eq.2.1. Then we validate the candidate depth value $z_c(v_i)$ and $\mathbf{n}_c$ using the differential image I_t with Eq.2.9. For each vertex, we choose the candidate depth value that minimizes the difference between the measured and modeled I_t, to be the initial depth. The complete near-light photometric stereo with circular placed LEDs is summarized in Algorithm.1.

Algorithm 1 Near-light Photometric Stereo with Circular Placed LEDs

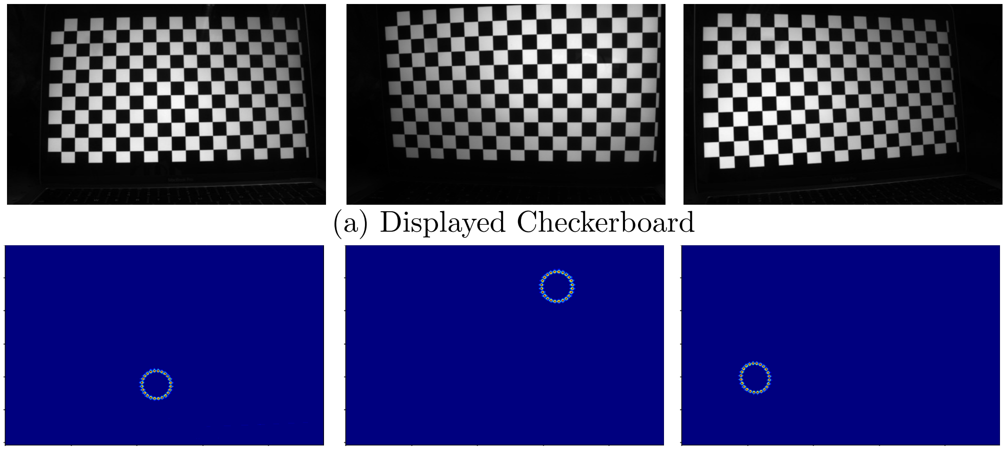
```
Given images $I$, differential images $I_t$, Camera Intrinsic Matrix $K$, Light Source Positions $S$
    and Light Source motion vectors $S_t$;
Initialize the depths with line search for each vertex.
Estimate the depths $z_{I_t}$ using Eq. 2.12 ;
Initialize the albedo $\tilde{\rho}_{I_t}$ given $z_{I_t}$;
Initialize: $z_I^{(0)} = z_{I_t},\ \tilde{\rho}^{(0)} = \tilde{\rho}_{I_t},\ k = 0$
for k $\in \{1, \cdots, \mathit{MaxIter}\}$ do
    Get $z_I^{(k)}$ using Eq.2.7, with $z_I^{(k-1)}$, $\tilde{\rho}^{(k-1)}$ as the initials
    Given $z_I^{(k)}$, solve for $\tilde{\rho}^{(k)}$ using Eq.2.6
end for
return $z_{opt} = z_I^{(k)}$, $\tilde{\rho}_{opt} = \tilde{\rho}^{(k)}$
```

2.5.1 Localizing Light Sources

It is important to calibrate the 3D light source positions accurately since we use the first-order derivative of the source positions. The calibration error introduced by traditional calibration methods using one or multiple chrome spheres will fail our algorithm, since the precise 3D location and projected radius of the sphere in the image plane required by these methods are difficult to measure or calibrate automatically. Instead, we propose a light source position calibration method using a flat specular display: First, we display the checkerboard pattern on a planar glossy display such as the monitor of a Macbook and capture one image for each setup of the plane, as shown in Fig.2.5(a). For each plane orientation, we turn off the display and turn on the LEDs sequentially and capture one image for each LED.

For each plane setup, we can get the plane parameters (plane orientation and distance from the origin) from the well-established camera calibration process [92]. Given the camera intrinsic matrix and plane parameters , for each light source reflection, we then estimate the light position by ray-tracing and triangulating for the centers of highlights in the light source reflection images shown in Fig.2.5(b).

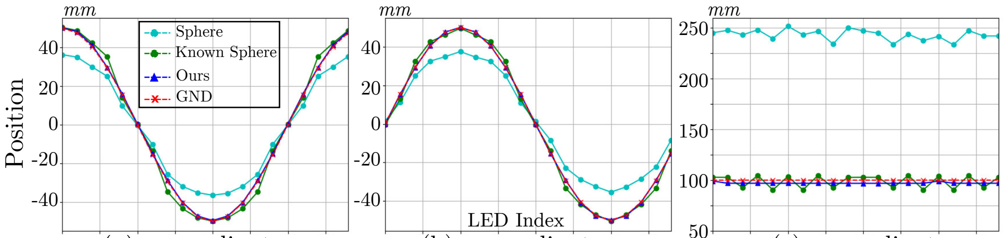
We evaluate the performance of our calibration procedure and compare it with the method using chrome spheres [148]. To demonstrate the sensitivity of the calculated light source locations *w.r.t.*



(a) Displayed Checkerboard

(b) Reflected Light Sources

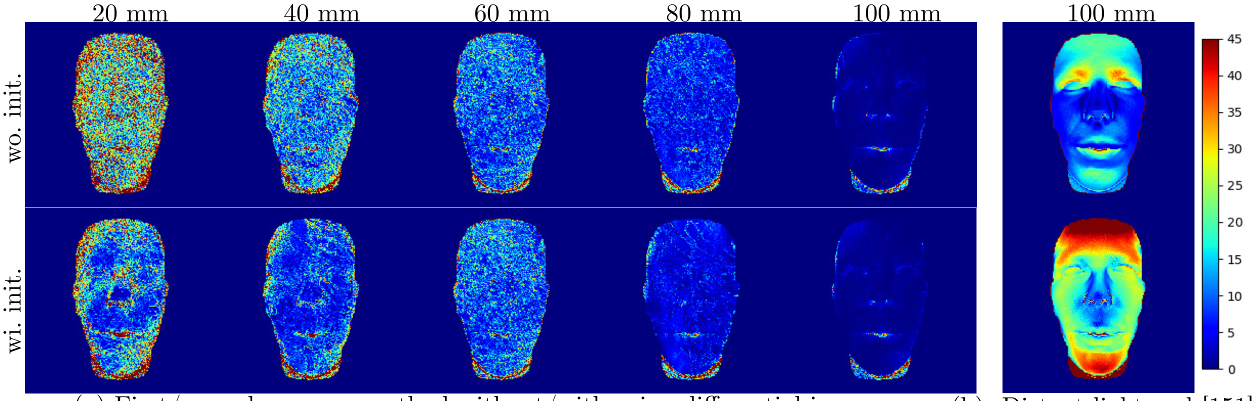
Figure 2.5: We calibrate the camera and the point light source positions using a planar glossy display (Macbook monitor). (a)Images captured with the display turned on and light source turned off, from which we estimate the camera intrinsic matrix and plane parameters. (b)Superposition of the images captured when the display is turned off and the light sources are turned on.



(a) x coordinate (b) y coordinate (c) z coordinate

Figure 2.6: The xyz profiles of the locations for all point light sources. Cyan - sphere chrome based method; The ellipse parameters are estimated using [145]; Green - sphere chrome based method; The ground truth location of the sphere center is given. Blue - Our method; Red - Ground truth.

the sphere center estimation, we evaluate the performance with/without using the ground truth 3D position of the sphere center. In the case where the ground truth 3D position is unknown, we use the ellipse detector in [145] to get the ellipse parameters. Then we approximate the ellipse to be round circle with center same as the ellipse center and radius as the mean of axes lengths of the ellipse. Given the physical dimension of the sphere and the camera focal length, we can get the depth and the location of the sphere center as in [148]. The comparison results in simulation are shown in Fig.2.6. As shown by the cyan curve in Fig.2.6, inaccurate estimations for the 3D locations of sphere centers lead to large errors, especially in the z direction. By contrast, our method is much more accurate.



(a) First/second row: our method without/with using differential images (b) Distant light and [151]

Figure 2.7: Ablation study for using the differential image I_t in Eq.2.12 with different LED ring radii. The mean depth for the object is 900 mm. For radius $= 100\ mm$, we compare our method with photometric stereo with distant light assumption, and the method in [151]. (a) Our method without and with the depth estimation using Eq.2.12 to initialize the depths for the optimization in Eq.2.7. Initialization using the differential images in Eq.2.12 helps to improve the reconstruction, especially for smaller light source baselines. For both setups, the error decreases with larger LED baselines. In our real experiment setup, we use an LED ring with a radius of 30 mm. (b) Error maps using distant light assumption (first row) and method in [151] (second row). Both methods result in large errors even for the largest LED baseline.

2.6 Experiments

2.6.1 Implementation Details

We implement the differential motion of the light source with an LED ring with 24 LEDs and 30 mm in radius. The reconstructed objects are placed 300 - $400\ mm$ away from the camera and light sources. We use the Prosilica GT1930c camera manufactured by Allied Vision to capture the images. Each image is captured with $.1$ second exposure time with one LED turned on. The algorithm is implemented in Python and C++, with the Ceres-Solver [5] for optimization. For the energy functions defined in Eq.2.7 and Eq.2.12, we set the weights for the smoothness term E_s to be $\lambda_I = .1$ and $\lambda_{I_t} = .01$ respectively. For faster convergence, we perform the optimizations in multiple scales where the results from lower resolution are used as the initializations for the higher resolution. The running time for 968×608 image resolution is about 5 min using on a desktop with Intel Core-i7 5940 CPU and 64 GB RAM memory size. We will release the implementation upon publishing.

2.6.2 Simulations Results

We test our algorithm with different imaging setups with synthesized images. To validate the effectiveness of the initialization using differential images, we place the reconstructed surface at

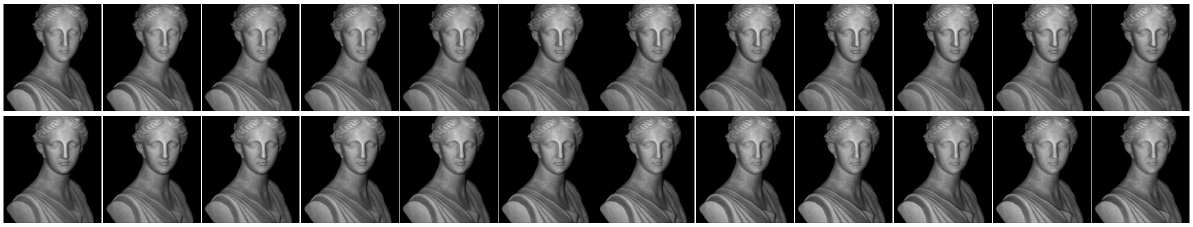
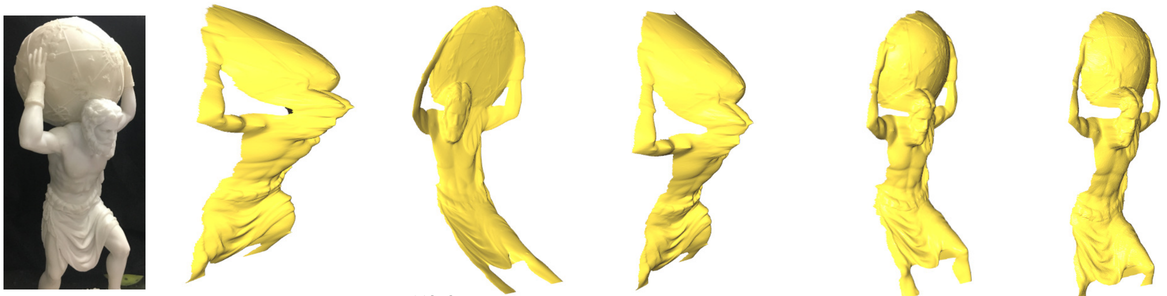


Figure 2.8: The input images with the light sources on the LED ring turned on sequentially. The radius of the LED ring is 30 mm and the object is placed around 400 mm away. The image pair shown in the same column corresponds to LED pairs on the opposite sides of the ring. As shown, even for the largest LED baselines (the image pair shown in one column), the difference between images is still small.



(a) Reconstructed Object (b)Distant light with $z_{\text{init}} = 200mm$ (c)[22] with $z_{\text{init}} = 200$ mm (d)Distant light with initialized z_{It} (e) [22] with initialized z_{It} (f) Our method

Figure 2.9: We show the effectiveness of the first-stage of results by showing the reconstruction for the atlas statue using different methods with/without initialization using z_{I_t}. The performance for both compared methods increases by using z_{I_t}. (a)Object profile; (b)Reconstruction with distant light source assumption; (c)Reconstruction using method in [151]; (d)(c)Reconstruction with the comparison methods, with z_{I_t} as the initialization; (f) Our method.

a plane with 900 mm depth, facing towards the camera. We reconstruct the surface with and without the first stage of the proposed method. For the compared method with no initializations, we set initial depth to be 200 mm. We run this comparison for multiple image settings where the LED ring radius ranges from 20 mm to 100 mm. Then we measure the angles between the estimated and ground truth surface normals to quantify the performance. As shown in Fig.2.7, the initialization using the differential images helps to improve the reconstruction, especially for smaller light source baselines. For larger LED ring radius, the performance of the method proposed in Sec.2.3 without initialization is comparable to the method in Sec.2.5 with depth initialization. This might be because there are less local minima for the energy function in Eq.2.7 for larger light source baselines. With the same 200 mm depth initialization, both photometric stereo using distance light source assumption and the method in [151] induce large errors even for the largest light source baseline.

We evaluate our method for 6, 10, 14 and 18 LEDs with the same scene setup and LED ring dimension. The mean surface normal error is $10.42, 3.15, 2.63$ and 2.56 degrees respectively. Note that the error sharply drops as we increase the LEDs from 6 to 10. This shows that the small light source baseline makes the approximation in Sec.2.4 valid even for fewer LEDs.

2.6.3 Real World Results

We apply our algorithm on images captured with LEDs on a ring with 30 mm in diameter. One sequence of capture images is shown in Fig.2.8 for a bust statue placed around 400 mm away from the camera. Due to the small ratio between the LED ring radius and the object-to-camera distance, the difference between images is very small even for the pair of LEDs with the largest baselines. Despite the small difference, our method still performs well as shown in Fig.2.10 where both the large depth variation between the left and right shoulders, and the fine grained structures in the frontal clothes are reconstructed. Note that for small baseline near-light photometric stereo, the optimization for depths is more easily trapped in the local minima due to the fact that the changes in the intensities are small. Thus using just the image intensity I is likely to generate degraded reconstructions if the initialization is not good, as shown both in Fig.2.3 with our image formation in Eq.2.1 , and in Fig.1.4(d). One extreme case would be that the baseline for the light source is zero. In this case the problem becomes the highly ill-posed shape from shading problem and can be solved only with prior-knowledge about the shape geometries [7, 209]. In our case, for near-light photometric stereo with small light source baselines, the initializations using I_t helps the optimization in Eq.2.7 process to avoid poor initializations and keep it from getting trapped in the surrounding local minima in the first stage.

To further validate the effectiveness of the first-stage of our method, we apply both traditional photometric stereo under distant lighting assumption and the method in [151] with the estimation results z_{It} as the initial depth values. For distant-light photometric stereo, we use those initialized depth values to get the lighting directions for all points; for the method in [151], we use z_{It} as its initial depth guess for optimization. As shown in Fig.2.9, by using z_{It}, the performance for both compared methods increases. Note in Fig.2.9(e), the reconstructed for the right leg of atlas is in accurate by bending forward, even though we have initialized the depth estimation process with I_{It} for this case.

We apply our method on other objects with different scene geometries. Results are shown in Fig.2.10. The first two columns of Fig.2.10 are two input images taken with lights on the opposite sides of the LED ring turned on. Although the image difference is small, our method is able to recover both the overall shape with enough depth variations, such as the shape of face, and fine-grained details, such as the logo on the tennis shoe.

For small light source position changes, we assume that the global component does not change much. As a result, we cancel out the global illumination component by subtracting two images captured with close light sources during estimating I_t. Based on this observation, we can further refine the reconstruction results by adding another step where the analytical form of I_t in Eq.2.9 is used for optimizing depths, with z_{opt} in Algorithm 1 as the initial values. We test this idea for reconstructing the object surface with large concavity such as the bowl shown in Fig.2.12. As shown, we get more robustness against the global illumination component by using the differential images I_t.

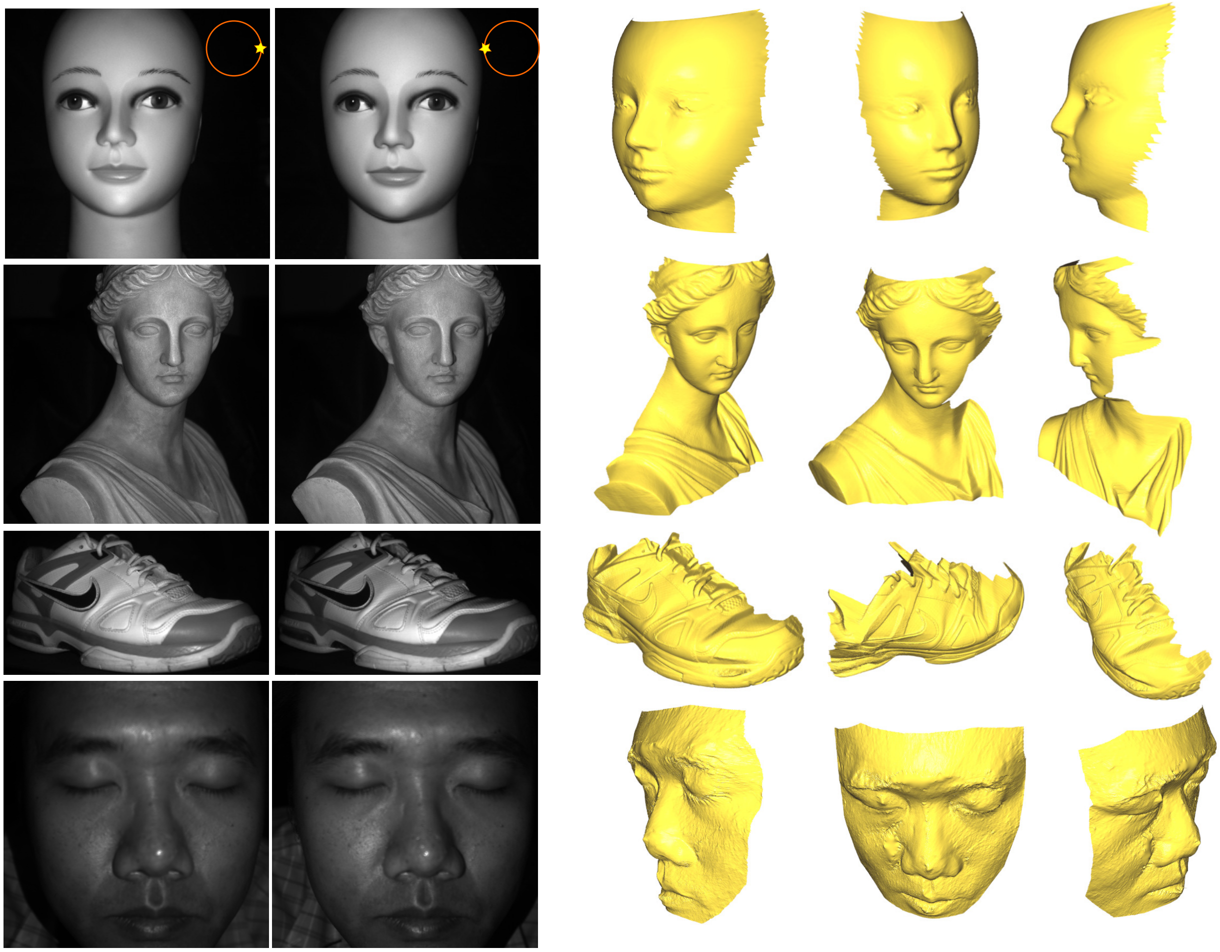
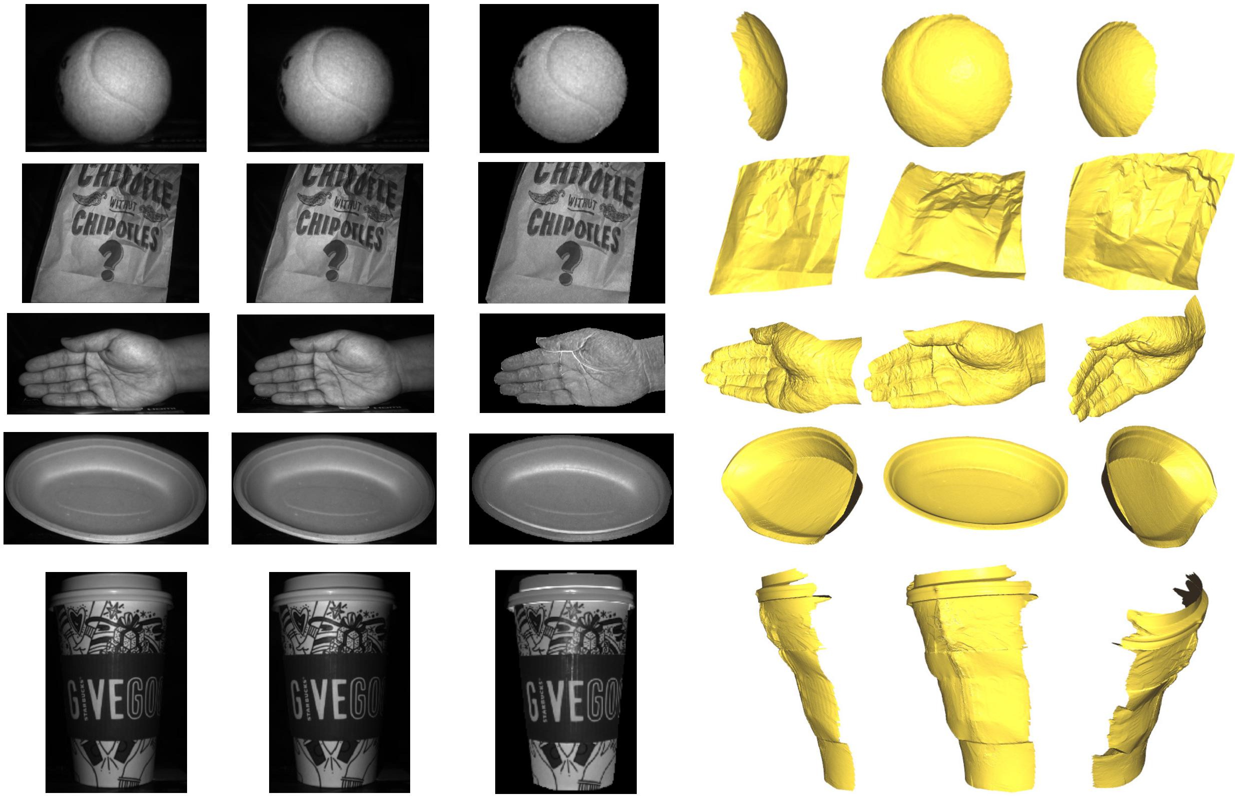


Figure 2.10: Inputs and reconstruction results using the proposed method. From left to right: Two of 24 input images taken with lights on the opposite sides of the LED ring turned on. Reconstructed meshes viewed from different views.

2.7 Limitations

The Lambertian assumption in our method fails when the surface includes specular reflection component, as can be seen in the human face reconstruction example in the last row of Fig.2.10. This leads to high-frequency artifacts such as the spike on the reconstructed nose. Another limitation of our method is that even though using the differential images leads to more robustness to the global light component as shown in Fig.2.12, the global component is not fully modeled and removed during reconstruction. So the reconstruction error in the presence of global component is still observable. One future direction is to include both the BRDF model and global illumination term into our problem formation.

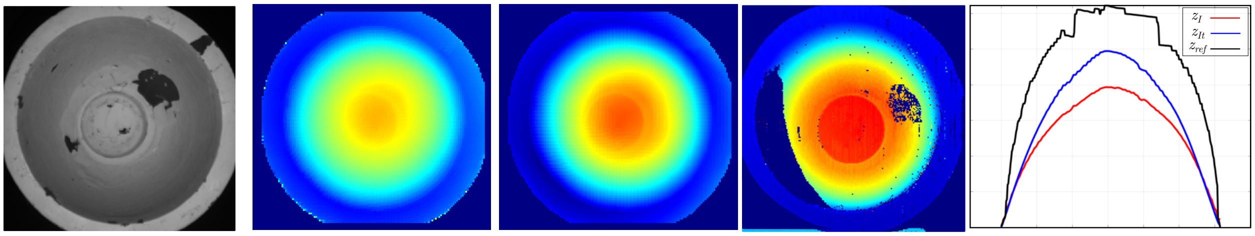


(a) Two of 24 input images (b) Albedo map (c) Surface reconstruction

Figure 2.11: Inputs and reconstruction results for the proposed method. (a) Two of 24 input images taken with lights on the opposite sides of the LED ring turned on; For each column, the LED positions are the same; (b) The estimated albedo map; (c) Reconstructed surfaces viewed from different angles.

2.8 Conclusion

In this chapter, we put forward a two-stage near-light photometric stereo algorithm with circularly placed point light sources and a pinhole camera. In the first-stage, we optimize the scene depth using the differential images captured by moving the light source slightly. We show in the chapter that the surface reconstruction becomes less non-linear by using the differential images. In the second stage, we refine the estimations using the raw captured images. We validate that our method is able to get good reconstruction results even with small baseline point light sources such as a low-cost LED ring. One future direction is to consider cases with general BRDFs and global illumination.



(a) One input image (b) Depth estimation using I (c) Depth estimation using I and It (d) Reference depth map using structured light (e) Depth profiles

Figure 2.12: We reconstruct the inner surface of a concave bowl. We get more robustness against the inter-reflection during reconstruction with the differential images I_t. From left to right: (a) one of the input images; (b) estimated depth map using the raw images I only; (c) estimated depth map using both I and differential images I_t; (d) reference depth map estimated using structured light with global-direct light separation; (e) 1D depth profiles for different methods (red: using I only; blue: using I and I_t; black: reference).

Chapter 3

Thin Structures Reconstruction using Single View Focal Stack

3.1 Introduction

Thin structures such as meshes, grass or tree branches are common in photography. Similarly, in medical and microscopic imaging, thin curvilinear structures such as vessels and neurons appear very often. Recovering the 3D information for such structures with non-invasive imaging modalities is useful for study of plants [54, 189], blood vessels [138, 175], and neurons [40, 84].

Segmenting thin structures from the background and recovering their depths is a challenging task for multiple reasons. First, thin structures located in close range might occlude more distant objects. So the ray corresponding to a pixel may encounter multiple occluders at different depths due to the partial occlusion. Second, the 3D structures of curvilinear objects in nature such as vessels and grass are often complex and non-planar, thus the methods based on planarity assumption [56, 73, 211] fail in those cases. Third, because of the small widths of the thin structures, the high spatial frequency depth discontinuities are likely to be recovered coarsely using patch-based depth-from-focus/defocus methods [49, 77, 78, 178].

In this work, we present a method for matting and depth recovery of 3D thin structures with self-occlusions using single-view focal stack images. To this end, we first propose a general image formation model that explicitly describes the spatially varying blur and multiple partial occlusions along a line of sight. Jointly optimizing the occlusion mattes and depths in the model is computationally intractable. We derive a Markov Chain Monte Carlo (MCMC) inference algorithm for the thin structure matting where the image/model update is directly and analytically computed. The analytic computation enables efficient updates of the model without re-rendering new images during the MCMC process, which makes the algorithm practical. The depths of thin structures are then recovered using gradient descent with the differential terms calculated from the model.

We evaluate the performance of the proposed method using images of scenes at both macro and micro scales. For macro-scale, we evaluate our method on scenes with complex 3D thin structures such as meshes, tree branches and grass. For micro-scale, we apply our method to in-vivo microscopic images of micro-vessels with diameters less than $50\ \mu m$. We reconstruct the 3D

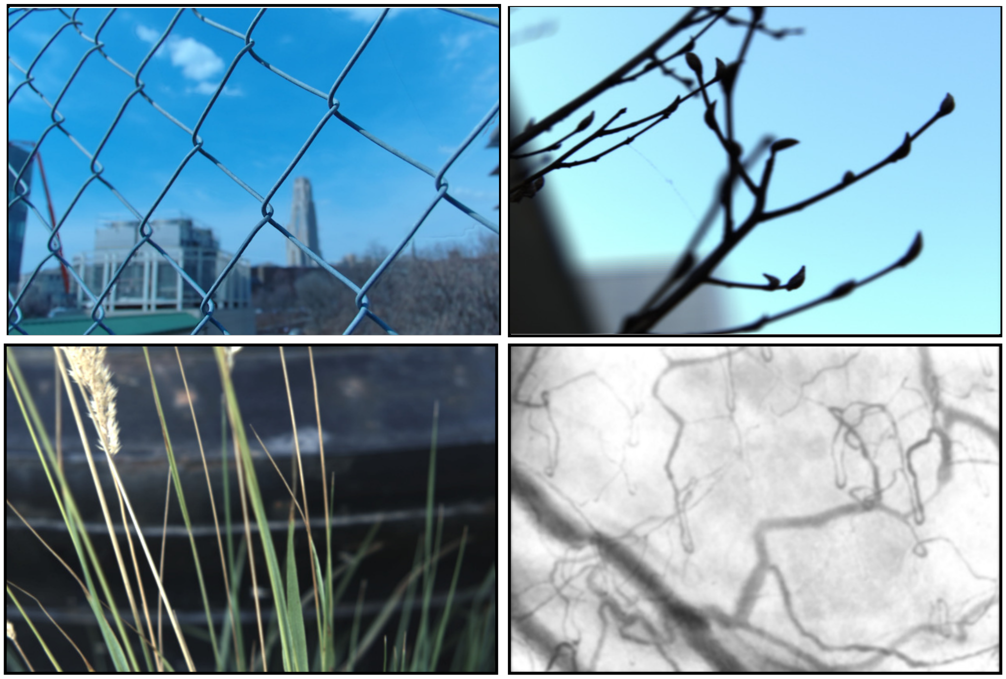


Figure 3.1: Example scenes with thin structures: mesh, grass, tree branches, and micro-vessels. Such structures are often non-planar, located at multiple depths, and occluding one another. The goal of this chapter is to matte and recover depths of these thin structures from a single-view focal image stack.

structure of the micro-vessels despite spatially varying blur and occlusions. To our knowledge, this is the first method to reconstruct the 3D structures of micro-vessels from a non-invasive in-vivo imaging system.

3.2 Related Work

Occlusion estimation and removal: Learning-based and physics-based methods have been used to remove occluders or recover the depths and patterns of the occlusions. In [51], a neural network was trained to detect and remove the dirt of rain drops. In [116] the translational symmetry pattern of the foreground has been exploited. Other methods estimate and remove the occlusion by using an image formation model that takes into account occlusions. [55, 73, 122]. In [55], an inverse projection model is used to recover the geometry and radiance of the scene following a variational framework. Gu *et.al* [73] model the captured radiance as a superposition of the foreground then recover the occlusion pattern and the occluded background from images captured with different focus settings by assuming that the foreground is fronto-parallel and dark. In [194], the occlusions are removed using large synthetic aperture images captured with an array of cameras.

Scene matting with obstructions: Xue *et.al* [211] exploit the difference between the edge flows of the obstruction surface and the background in a video to separate and recover the foreground and background radiances. In [56], light field matting is used to recover both the foreground and background layers. In [77, 78], the simplified multilayer scene model, where the radiance is assumed to come from an all-in focus scene layer, is solved in order to perform post-capture image refocus. The radiance for all layers are approximated by a single all-in-focus radiance map. For thin structure occlusions in [56], the multilayer model is simplified to consist of a single pair of fronto-parallel foreground and background layers. Rather than first simplify the multilayer model and then solve the more constrained problem like in [77] and [56], we will directly solve the full multilayer model with multiple non-fronto-parallel occlusion layers.

Reconstruction and depth estimation with occlusions: Due to lack of correspondences, traditional 3D reconstruction methods such depth from defocus and stereo matching fail to work well on scenes with occlusions. Yamazaki *et.al* in [212] use shadows cast from a point light source to reconstruct intricate objects that are difficult for traditional shape-from-silhouettes methods.

In [201], the occlusions have been modeled in the 4D light field and the occlusions are explicitly handled to get better depth estimation near depth disparities. Photo-consistency is extended to points at the depth disparity edges to handle occlusions more explicitly. The partial occlusion is modeled in the angular space of the input 4D light field. In our method, the occlusions are modeled using the multilayer matting function based on 2D spatially varying defocus kernels. In addition, we also demonstrate our approach in cases where occluders block each other.

In [73], a single fronto-parallel layer of occlusions is removed using two or three images captured with different aperture sizes. The occlusions are assumed to be dark without contributing any radiance. In [56], the occlusion is also assumed to be in single fronto-parallel layer. In contrast, we address occlusions that are located in different depths and may occlude one another.

3.3 Image Formation Model

For a camera with finite aperture, one pixel at the image plane receives radiance contributions from *multiple* rays from the points within a double-sided cone determined by the focal plane and aperture size, as shown in Figure 3.2. With the image coordinate denoted as $\mathbf{v}$, we represent the occluder with occlusion matte $M(\mathbf{v}) \in \{0, 1\}$ and radiance $L(\mathbf{v}) \in \mathcal{R}^+$. If there is only one opaque occluder in the scene, the image intensity at $\mathbf{v}$ in the m-th image R^m in the focal stack is

$$R^m(\mathbf{v}) = \int_{\mathbf{u}} L(\mathbf{u}) M(\mathbf{u}) B^m\left(\mathbf{v} - \mathbf{u}; d(\mathbf{v})\right) \mathrm{d}\mathbf{u}$$

where $B^m\left(\mathbf{v} - \mathbf{u}; d(\mathbf{v})\right)$ is the spatially varying blur kernel dependent on the scene point depth $d(\mathbf{v})$.

For scenes with opaque occluders located at multiple depths, the image intensity for one pixel is contributed by multiple points at different depths, with possible attenuations due to occlusions as shown in Figure 3.2. We denote the occlusion index $k \in \{1, 2, \cdots, N\}$ to be the order in which the double-sided cone from the camera encounters the scene points. The image R^m is the superposition of contributions from scene points across all occlusion indexes:

$$R^m(\mathbf{v}) = \sum_{k=1}^{N} \alpha_k^m(\mathbf{v}) \int_{\mathbf{u}} L_k(\mathbf{u}) M_k(\mathbf{u}) B_k^m\left(\mathbf{v} - \mathbf{u}; d_k(\mathbf{v})\right) \mathrm{d}\mathbf{u} \tag{3.1}$$

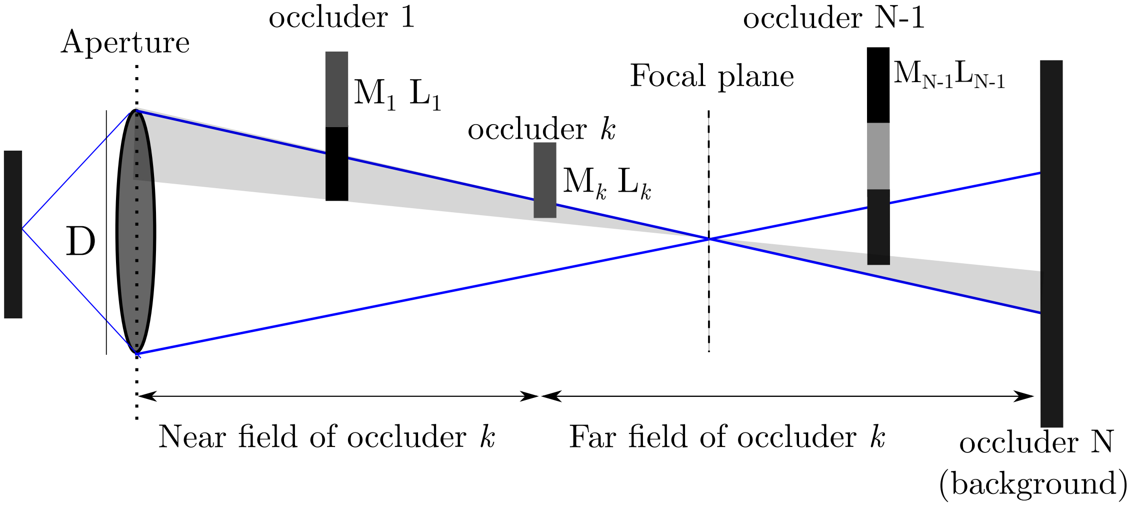


Figure 3.2: Viewing geometry of a single pixel in a camera with finite aperture. The camera is focused between occluder k and occluder $N-1$. The pixel receives radiance contributions from rays within the double-sided cone determined by the focal plane and aperture size. The occluders are represented with the occlusion map M and radiance map L. Occluder k is partially occluded by the occluders in its near field and occludes the occluders/background in its far field.

with $B_k^m(\mathbf{v}-\mathbf{u};d_k(\mathbf{v}))$ denoting the spatially varying blur kernel for the scene point with occlusion index k. The attenuation term $\alpha_k^m(\mathbf{v})$ describes the attenuation of the radiance from occluder k due to occlusions. As shown in Figure 3.2, the occluder with occlusion index $k > 1$ is only obstructed by points in the near field with occlusion index smaller than k, thus the attenuation term can be written as:

$$\alpha_k^m(\mathbf{v}) = \begin{cases} 1, & \text{if } k = 1 \\ \prod_{j=1}^{k-1} 1 - \int_{\mathbf{u}} M_j(\mathbf{u}) B_j^m(\mathbf{v}-\mathbf{u}; d_j(\mathbf{u})), & \text{otherwise} \end{cases} \tag{3.2}$$

Eq. 3.1 and Eq. 3.2 describes the general case shown in Figure 3.2 where the defocus blur is spatially-variant and the occluders in the scene may partially occlude one another.

Because the blur kernels in Eq. 3.1 and Eq. 3.2 are compact in space, the range of $\mathbf{u}$ in the integral is within a local patch $\mathcal{N}(\mathbf{v})$. So we can write the discretized image formation model as:

$$R^m(\mathbf{v}) = \sum_{k=1}^{N} \alpha_k^m(\mathbf{v}) \sum_{\mathbf{u}\in\mathcal{N}(\mathbf{v})} L_k(\mathbf{u}) M_k(\mathbf{u}) B_k^m\left(d_k(\mathbf{v})\right) \tag{3.3}$$

with the discretized attenuation term:

$$\alpha_k^m(\mathbf{v}) = \begin{cases} 1, & \text{if } k = 1 \\ \prod_{j=1}^{k-1} 1 - \sum_{\mathbf{u}\in\mathcal{N}(\mathbf{v})} M_j(\mathbf{u}) B_j^m(d_j(\mathbf{u})), & \text{otherwise} \end{cases} \tag{3.4}$$

with $B(d(\mathbf{u})) = B(\mathbf{v} - \mathbf{u}; d(\mathbf{u}))$ for notation simplicity.

The image formation model in Eq. 3.3 and Eq. 3.4 generalizes the models used in previous works. When occluders are fronto-parallel, the blur kernel for each occluder is spatially-invariant. In this case the integrals in Eq. 3.3 and Eq. 3.4 become convolutions with blur kernels $B_j^m(\mathbf{v} - \mathbf{u})$. For $N = 1$, and the image formation model becomes:

$$R^m(\mathbf{v}) = L_1 M_1 * B_1(\mathbf{v})$$

which is the scene model used in [77, 78] for depth recovery and post-capture re-focusing. For $N = 2$, there is only one occlusion in front of the background, the image formation model becomes:

$$R^m(\mathbf{v}) = L_1 M_1 * B_1(\mathbf{v}) + (1 - M_1 * B_1(\mathbf{v}))(L_2 * B_2(\mathbf{v}))$$

which is the image formation model used in previous works on image matting [105, 123] and occlusion reasoning [73].

3.4 Efficient MCMC for Occlusion Matting

In this work, the goal is to estimate the occlusion matte $M_k(\mathbf{v})$, depth $d_k(\mathbf{v})$ and scene radiance $L_k(\mathbf{v})$ for occlusion index $k \in \{1, 2, \cdots, N\}$, given the measured focal stack images and calibrated defocus blur kernels B_k^m. In the following, we will first describe our method to estimate the occlusion mattes from a focal stack, followed by the depth recovery for the occluders explained in Section 3.5.

Given the measured focal stack images $\{I^m(\mathbf{v})\}$ captured with different focal plane distances, the estimated occlusion mattes $M(\mathbf{v})$ are determined by minimizing the energy function:

$$E(M(\mathbf{v})) = E_{\text{data}}(M(\mathbf{v})) + \lambda E_{\text{smooth}}(M(\mathbf{v}))$$

with

$$E_{\text{data}} = \sum_{m,\mathbf{v}} (I^m(\mathbf{v}) - R^m(\mathbf{v}))^2$$

$$E_{\text{smooth}} = \sum_{(\mathbf{u},\mathbf{v}) \in \mathcal{N}_8} 1 - \delta(M(\mathbf{u}) - M(\mathbf{v}))$$

where the smoothness term E_{smooth} enforces the local spatial consistency for occlusion matting. $R^m(\mathbf{v})$ is the forward rendered image using the image formation model in Eq. 3.3 and Eq. 3.4. We can see from Eq. 3.4 that changing the occlusion matte value $M_j(\mathbf{v})$ will effect the attenuation terms α_k for all $k > j$. The range of the influence is the size of the blur kernel, which could be large when the occluder is highly defocused. This influence is propagated to the other occlusion mattings M_k through Eq. 3.3. Therefore, there are high-order relationships among the occlusion mattings values. So the data term E_{data} is of high-order w.r.t. $M_k(\mathbf{v})$ for $k \in \{1, 2, \cdots, N\}$.

Because of these high-order relationships, traditional graph-based methods dealing with relatively low-order potentials will not apply. Methods that include high-order potentials [57, 89, 99, 181] either require the graph to be in specific structure [181] or the relationship can be analytically modeled [57, 89, 99]. Instead, we derive an efficient MCMC inference method where the image/model updates are *directly* and *analytically* computed based on the image formation model without re-rendering the images. This makes an otherwise intractable problem practical to solve.

We will assume: 1) the radiances of the thin structures are different from the radiance of background; 2) the maximal number of occlusions along a line of sight is known or pre-set. The first assumption enables us to detect and separate the occluders from background using the focal stack; the second assumption simplifies the derivation.

MCMC inference:

Consider a point $\mathbf{x}$ on the k-th occluder on the line of sight, as shown in Figure 3.2. During the MCMC inference process, the occlusion matte value $M_k(\mathbf{x}) \in \{0, 1\}$ is sampled from the probability distribution:

$$\begin{aligned} p\left(M_k(\mathbf{x}) = 1\right) &= \frac{\mathrm{e}^{-E(M_k(\mathbf{x})=1)/T}}{\mathrm{e}^{-E(M_k(\mathbf{x})=1)/T} + \mathrm{e}^{-E(-M_k(\mathbf{x})=0)/T}} \\ &= \frac{\mathrm{e}^{-\Delta E(\mathbf{x})/T}}{1 + \mathrm{e}^{-\Delta E(\mathbf{x})/T}} \end{aligned} \tag{3.5}$$

where $E(M_k(\mathbf{x}) = b)$ for $b = \{0, 1\}$ are the energy functions for the binary assignments for $M_k(\mathbf{x})$; $\Delta E(\mathbf{x}) = E(M_k(\mathbf{x}) = 1) - E(M_k(\mathbf{x}) = 0) = \Delta E_{\text{data}} + \lambda \Delta E_{\text{smooth}}$ represents the increase of the energy function when the sampling in the MCMC process changes the occlusion matte value at $\mathbf{x}$ $M_k(\mathbf{x})$ from 0 to 1. T is the temperature parameter controlling the acceptance rate for an update and the convergence of the MCMC process.

Estimating ΔE_{data} for MCMC Inference:

By denoting $R^m(\mathbf{v}; b)$ to be the forward rendered image when $M_k(\mathbf{v}) = b$ for $b = \{0, 1\}$, the data term of ΔE can be written as:

$$\begin{aligned} \Delta E_{\text{data}} &= \sum_{m,\mathbf{v}} (I^m(\mathbf{v}) - R^m(\mathbf{v}; 1))^2 - (I^m(\mathbf{v}) - R^m(\mathbf{v}; 0))^2 \\ &= \sum_{m,\mathbf{v}} \Delta R^m(\mathbf{v}) \left(\Delta R^m(\mathbf{v}) + 2\left(R^m(\mathbf{v}; 0) - I^m(\mathbf{v})\right)\right) \end{aligned} \tag{3.6}$$

where $\Delta R^m(\mathbf{v}) = R^m(\mathbf{v}; 1) - R^m(\mathbf{v}; 0)$ is the change of the rendered image by changing $M_k(\mathbf{v})$ from 0 to 1. Similarly, we can write the change of the data term for switching $M_k(\mathbf{v})$ from 1 to 0 as:

$$\Delta E_{\text{data}} = \sum_{m,\mathbf{v}} \Delta R^m(\mathbf{v}) \left(-\Delta R^m(\mathbf{v}) + 2\left(R^m(\mathbf{v}; 1) - I_i(\mathbf{v})\right)\right) \tag{3.7}$$

Analytically Computing $\Delta R^m(\mathbf{v})$ for ΔE_{data}:

The naive approach is to render images $R^m(\mathbf{v}; 0)$ and $R^m(\mathbf{v}; 1)$ directly and estimate $\Delta R^m(\mathbf{v})$ for all pixels and occlusion indexes. In addition, we need several iterations since the results from the burn-in period of the MCMC process is not reliable. So the computational complexity for the naive approach is too high for any real world application.

Rather than perform the full forward render process for each pixel, we propose to directly and analytically compute the $\Delta R^m(\mathbf{v})$ and its corresponding energy difference ΔE_{data} by using the image formation model in Eq. 3.3 and Eq. 3.4. The image intensity change $\Delta R^m(\mathbf{v})$ induced by switching $M_k(\mathbf{x})$ from 0 to 1 is contributed by radiance change from occluder k and occluders with occlusion index $i > k$ on the line of sight:

$$\Delta R^m(\mathbf{v}) = \alpha_k(\mathbf{v}) B_k^m(d_k(\mathbf{x})) L_k(\mathbf{x}) + \sum_{i=k+1}^{N} \Delta\alpha_i(\mathbf{v}) \tilde{L}_i(\mathbf{v}) \tag{3.8}$$

with the defocused image

$$\tilde{L}_i(\mathbf{v}) = \sum_{\mathbf{u} \in \mathcal{N}(\mathbf{v})} L_i(\mathbf{u}) M_i(\mathbf{u}) B_i^m(d_i(\mathbf{u})), \tag{3.9}$$

where $B_k(d_k(\mathbf{x}))$ and $B_i(d_i(\mathbf{u}))$ are spatially varying blur kernels; $\Delta\alpha_i(\mathbf{v})$ is the change of attenuation by switching the occlusion matte value $M_k(\mathbf{x})$ from 0 to 1. The first term in Eq. 3.8 is the radiance change contribution from the k-th occluder. The second term is the radiance change contributions from the occluders/background in the far field of occluder k.

Analytically Computing $\Delta\alpha_i(\mathbf{v})$ for $\Delta R^m(\mathbf{v})$:

For notation simplicity, we denote the blurred occlusion matte in Eq. 3.4 with:

$$\tilde{M}_j(\mathbf{v}) = \sum_{\mathbf{u} \in \mathcal{N}(\mathbf{v})} M_j(\mathbf{u}) B_j^m(d_j(\mathbf{u})) \tag{3.10}$$

From Eq.3.4, the attenuation change $\Delta\alpha_i(\mathbf{v})$ can be written as:

$$\begin{aligned} \Delta\alpha_i(\mathbf{v}) &= \Big(1 - \tilde{M}_k(\mathbf{v}; 1)\Big) \prod_{j=1; j\neq k}^{i-1} \Big(1 - \tilde{M}_j(\mathbf{v})\Big) \\ &\quad - \Big(1 - \tilde{M}_k(\mathbf{v}; 0)\Big) \prod_{j=1; j\neq k}^{i-1} \Big(1 - \tilde{M}_j(\mathbf{v})\Big) \\ &= -B_k(d_k(\mathbf{x})) \prod_{j=1; j\neq k}^{i-1} (1 - \tilde{M}_j(\mathbf{v})) \end{aligned} \tag{3.11}$$

By combining Eq. 3.8 and Eq. 3.11, we see that the image intensity change $\Delta R^m(\mathbf{v})$ induced by switching $M_k(\mathbf{x})$ from 0 to 1 is *independent* from $M_k(\mathbf{v}) \ \ \forall\ \mathbf{v}$. Before the MCMC process for points at occluders with occlusion index k, we can pre-compute $\tilde{L}_i$ in Eq. 3.9 and blurred occlusion mattes $\tilde{M}_j$ in Eq. 3.10. Then during the MCMC process, the image update $\Delta R^m(\mathbf{v})$ can be *directly* and *analytically* estimated from Eq. 3.8 and Eq. 3.11. If the occlusion matte at pixel $\mathbf{x}$ changes after sampling from Eq. 3.5, the updated image can be easily computed with $R^m(\mathbf{v}) \leftarrow R^m(\mathbf{v}) + \Delta R^m(\mathbf{v})$ without re-rendering the images. In addition, due to the limited size of the blur kernel, the spatial range of $\Delta R(\mathbf{v})$ is limited within a small patch $\mathcal{N}(\mathbf{x})$ rather than over the whole image. In our implementation, we choose the size of the patch $\mathcal{N}(\mathbf{x})$ to be 31-by-31. Therefore, the high-order data term change ΔE_{data} can be computed efficiently.

Estimating ΔE_{smooth}:

For the smoothness term change ΔE_{smooth}, since it does not include the forward rendering, it can be simply computed as:

$$\Delta E_{\text{smooth}} = \sum_{\mathbf{u} \in \mathcal{N}_8(\mathbf{x})} \delta(M(\mathbf{u})) - \delta(M(\mathbf{u}) - 1) \tag{3.12}$$

when $M(\mathbf{x})$ changes from 0 to 1 and

$$\Delta E_{\text{smooth}} = \sum_{\mathbf{u} \in \mathcal{N}_8(\mathbf{x})} \delta(M(\mathbf{u}) - 1) - \delta(M(\mathbf{u})) \tag{3.13}$$

for $M(\mathbf{x})$ changes from 1 to 0. $\mathcal{N}_8(\mathbf{x})$ is the 8-connectivity neighborhood of $\mathbf{x}$. As we can see from Eq. 3.12 and Eq. 3.13 the change of the smoothness term is simply the difference of numbers of occupant and empty pixels around $\mathbf{x}$.

Initialization:

A good initialization of the variables is important given the huge search space for the occlusion matte. To initialize the occlusion matte, for each pixel $\mathbf{v}$ in the measured image $I^m(\mathbf{v})$, we first compute the variance of Laplacian in the Lab color space of a local 9-by-9 patch around $\mathbf{v}$. For the occlusion matting M_k with occlusion index $k < N$, we set $M_k(\mathbf{v}) = 1$ if the maximal local variance happens in a focal depth is smaller than a pre-defined threshold and 0 otherwise. The matting $M_N(\mathbf{v}) = 1$ for all pixels for the background since any line of sight will intersect with the background. For depth initialization, the initial depth for the thin structures at one pixel is estimated as the depth index in the focal stack with the largest variance of Laplacian of a local patch around that pixel. The radiance for the points on the thin structures is the measured image intensity in the corresponding image in the focal stack. During the optimization, the radiance values are updated based on the current depth estimation, which is explained in the next section. Given the initialization, the steps for the MCMC inference for $M_k(\mathbf{v})$ are described in Alg. 2.

3.5 Estimating Depths of Thin Structures

In order to compute the depth, we assume that the objects are locally planar within a small area. Given the matting estimation, we first over-segment the matted thin structures into super-pixels using SLIC [4] implemented in [196]. To get small and thin super-pixels, we set the area of the super-pixel to be 10 and the regularization factor to be 0.1. Each super-pixel will be treated as one tiny planar segment in space. The depth of the occluder is recovered by optimizing the parameters of all the foreground planar segments such that the synthetic images given the depth are as close as possible to the measured focal stack.

Given a planar segment i with plane parameters $\mathbf{s}_i$, the depth of the point on the segment with pixel coordinate (x, y) is $d = \mathbf{s}_i^T(x, y, 1)$. By concatenating all the plane parameters for N_s

Algorithm 2 Efficient MCMC inference for occlusion matte $M_k(\mathbf{v})$

```
Given initialization of $M_k^{(0)}$, $d_k$ and $L_k$ render $R^{(0)}$
for each iteration $t$ do
    for each occlusion index $k \in \{1, 2, \cdots, N-1\}$ do
        compute $\alpha_k(\mathbf{v})$ and $R$ using Eq.3.3 and Eq.3.4
        update $\tilde{M}$ and $\tilde{L}$ using Eq.3.10 and Eq.3.9
        for each pixel $\mathbf{x}$ with occlusion indx $k$ do
            compute $\Delta R$ using Eq. 3.8 and Eq. 3.11
            compute $\Delta E_{\text{data}}$ using Eq. 3.6 or Eq. 3.7;
            compute $\Delta E_{\text{smooth}}$ using Eq. 3.12 or Eq. 3.13;
            sample $M_k(\mathbf{x})$ using Eq. 3.5;
            $R \leftarrow R + \Delta R$ if $M_k(\mathbf{x})$ changes.
        end for
    end for
end for
```

segments into a $3N_s$-dimensional vector s, the optimal parameters for segment planes are found by:

$$\min_{\mathbf{s}}. \sum_{n,m,\mathbf{v}} (I_n^m(\mathbf{v}) - R_n^m(\mathbf{v};\mathbf{s}))^2 + \lambda_d E_{\text{s}}(\mathbf{s}), \tag{3.14}$$

where $I_n^m(\mathbf{v})$ is the measured image intensity of segment n at pixel $\mathbf{v}$. The first term of the energy is the data term measuring the difference between the synthesized images and the measured focal stack. The second term $E_{\text{s}}(\mathbf{s})$ is the smoothness term enforcing the depth smoothness for adjacent segments in 3D space. For two adjacent segments representing by their plan parameters $\mathbf{s_i}$ and $\mathbf{s_j}$, the depth smoothness energy is defined as the depth difference for the pixels on their shared boundary:

$$\begin{aligned} E_s^{(i,j)} &= (\mathbf{d_i} - \mathbf{d_j})^T(\mathbf{d_i} - \mathbf{d_j}) \\ &= (\mathbf{s_i} - \mathbf{s_j})P_b^T P_b(\mathbf{s_i} - \mathbf{s_j}) \\ &= (\mathbf{s_i} - \mathbf{s_j})A^{(i,j)}(\mathbf{s_i} - \mathbf{s_j}), \end{aligned}$$

where the three-column matrix P_b consists of the homogeneous coordinates of the pixels on the boundary of segment i and segment j; $A^{(i,j)} = P_b^T P_b$. The smoothness energy for all pairs of adjacent segments can be written in a way such that it is quadratic in terms of the concatenated plan parameters s:

$$E_s(\mathbf{s}) = \mathbf{s}^T \Lambda \mathbf{s} = \mathbf{s}^T \sum_{(i,j)\in\mathcal{N}} \Lambda^{(i,j)}\, \mathbf{s}, \tag{3.15}$$

where $\Lambda^{(i,j)}$ is a $3N$-by-$3N$ sparse matrix for the neighborhood segments $\mathbf{s_i}$ and $\mathbf{s_j}$ with non-zero block entries $\Lambda_{i,i}^{(i,j)} = \Lambda_{j,j}^{(i,j)} = A^{(i,j)}$ and $\Lambda_{i,j}^{(i,j)} = \Lambda_{j,i}^{(i,j)} = -A^{(i,j)}$.

To optimize the objective function defined in Eq. 3.14 using gradient-based method, we also need to calculate the gradient of the data term with respect to the plane parameters s, for which we need to estimate:

$$\frac{\partial R}{\partial \mathbf{s}} = \frac{\partial R}{\partial \mathbf{d}}\frac{\partial \mathbf{d}}{\partial \mathbf{s}} = \frac{\partial R}{\partial \mathbf{d}} P \tag{3.16}$$

with P the N-by-3 location matrix with each row as the homogeneous coordinate $(x, y, 1)$ for one pixel.

For a point corresponding to pixel $\mathbf{v}$ on the k-th occluder, the gradient of the rendered image w.r.t. its depth can be written as :

$$\frac{\partial R}{\partial d_k}(\mathbf{v}) = \alpha_k(\mathbf{v}) L_k(\mathbf{v}) \frac{\partial B_k}{\partial d_k} + \sum_{i=k+1}^{N} \frac{\partial \alpha_i}{\partial d_i}(\mathbf{v}) \tilde{L}_i(\mathbf{v}) \tag{3.17}$$

with

$$\frac{\partial \alpha_i}{\partial d_i}(\mathbf{v}) = -\frac{\partial B_i}{\partial d_i} \prod_{\substack{j=1 \\ j \neq k}}^{i-1} 1 - \sum_{\mathbf{u} \in \mathcal{N}(\mathbf{v})} M_j(\mathbf{u}) B_j^m(d_j(\mathbf{u})) \tag{3.18}$$

The derivation is similar as in Section 3.4. The differential blur kernel $\frac{\partial B}{\partial d}$ is pre-computed during the calibration process. The gradient of Eq.3.14 can be evaluated by combing Eq.3.15, Eq.3.16 and Eq.3.17. We use the conjugate-gradient method for optimizing the plane parameters $\mathbf{s}$. Given the the optimal $\mathbf{s}$, the depth of the segments is calculated as $\mathbf{d} = P\mathbf{s}$.

3.6 Experiments

3.6.1 Implementation Details

For all experiments, we choose the size of the local patch for MCMC update to be 31-by-31. We set the maximal occlusion index $N = 3$. The temperature parameter T in Eq. 3.5 is set to 5 and the smoothness parameter λ in Eq. 3.5 is set to 0.8. For depth estimation, we set the depth smoothness factor λ_d in Eq. 3.14 to be 0.5 and the step size of the gradient descent to be 0.1. The MCMC process converges within 10 iterations and the gradient descent for depth recovery converges within 50 iterations. The running time on a 620x480 focal stack with 26 focal planes is about 20 min using MATLAB implementation on a desktop with Intel Core-i7 5940 CPU and 64 GB RAM memory size.

3.6.2 Calibrating Blur Kernels

For macro-scale scenes, we use a Lytro ILLUM light field camera to generate the focal stack with 26 focal planes. Using a light field camera avoids the magnification variation due to focal changes and the need for post-processing to compensate the magnification. The refocused images are estimated from the 4D light field images by shearing the light field and projecting it into 2D slices as described in [136].

We calibrate the blur kernels for a set of 21 reference depths from 200 mm to 1000 mm equally spaced with 40 mm. In the calibration process, we use a planar reference plane with checkerboard

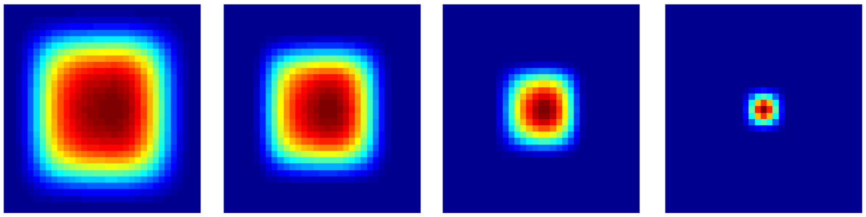


Figure 3.3: The calibrated blur kernels of refocused image for a plane placed 520 mm from the light field camera. The shapes of the blur kernels are not circularly symmetrical since the blur kernel for a refocused image from light field camera is related to both the main lens shape and the spatial arrangement of the secondary lenslets.

textures and place the plane parallel to the image plane. The optical blur kernel is assumed to be a separable filter kernel such that it can be written as a convolution of two 1D functions. Then the 1D functions are optimized.

Examples of the calibrated blur kernels for the focal stack images generated using the light field are shown in Figure 3.3. Note that the shape of the blur kernel is not circularly symmetrical since the blur kernel for a refocused image from light field camera is related to both the main lens shape and the arrangement of the secondary lenslets array. For the microscopic camera, we model the blur kernels as Gaussian functions with σ related to the focal plane distance and scene depth.

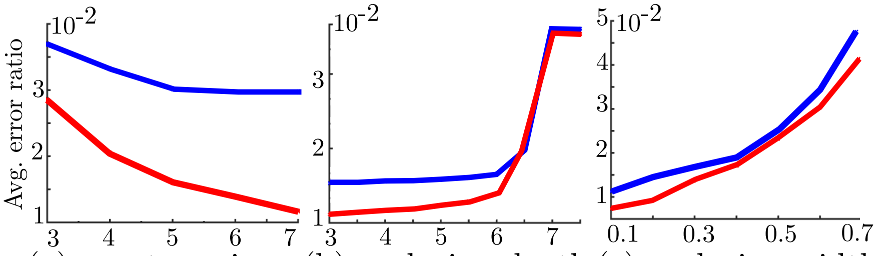
3.6.3 Aperture Size vs. Depth vs. Occluder Size

We first analyze the performance of our method under varying camera and scene configurations to evaluate the influence of aperture size, the depths and widths of the occluders. We synthesize the focal stack images with different camera and scene settings. With larger aperture size, we are able to collect rays from more angles coming from a point thus more rays can be imaged from the occluded regions [194]. The benefit of having a finite aperture decreases as the foreground occlusions are further from the camera. The synthetic scene includes two foreground occlusion layers with parts of the second layer being occluded.

The performance is evaluated in terms of the averaged error ratio of the rendered focal stacks. As shown in Figure 3.4, the reconstruction error decreases as the aperture size becomes larger since for larger aperture size, more rays from the partially occluded regions are collected. On the other hand, more reconstruction error of the background is introduced when the occlusion is closer to the background as regions in the background are completely occluded. Similar results can be observed for occlusions with different sizes.

3.6.4 Performance on Real Data

To quantitatively assess the performance of the proposed matting and depth recovery method, we place slanted planar mesh at measured distances and evaluate the matting and depth estimations. The selected set distances are listed in Table 3.1. We compare our method with the baseline depth-from-focus (DFF) method used in [183]. The occlusion matte is estimated by thresholding the



(a) aperture size (b) occlusion depth (c) occlusion width

Figure 3.4: The reconstruction error varies with camera aperture size, the depth and size of occluder. The blue and red curves in (a) are errors with occlusion distance set to 5 and 6 respectively. The blue and red curves in (b)(c) correspond to aperture size 5 and 11 respectively.

Table 3.1: RMSEs of the recovered depth for the slanted plane placed at different depths.

	Distance from the camera (mm)			
	250	380	510	680
DFF [183]	94.22	61.50	129.1	161.2
Proposed	30.49	34.35	36.13	60.18

recovered depth map based on the fact that the mesh plane is located in the near field. Because the degree of in-focus is measured from the image intensities within a local patch, the DFF method tends to generate an over-smoothed depth map where the depth estimations near the occlusion boundaries are inaccurate. In our method, since the defocus and occlusion are modeled explicitly for each pixel, we are able to recover the high frequency depth discontinuities for thin structures. Therefore, as shown in Table 3.1, the RMSEs of estimated depths for our method are lower than the ones for the DFF approach.

We also compare with the approach [201] using light field inputs with the occlusion boundaries explicitly modeled. As shown in Figure 3.5, the DFF method in [183] fails to recover high frequency depth changes in regions such as the edges of the grass where multiple depth discontinuities are close. This is because the patch-based estimation of the degree of in-focus will include the edges of the depth boundary even if the center of the patch is not aligned on the boundary. As a result, the degree of in-focus is inaccurate near the depth boundary. The method in [201] is able to estimate the depths at places where the occlusion boundaries are close because in this method the occlusion boundaries are modeled and processed explicitly. However, the approach in [201] is unreliable for textures regions in the background. In addition, some sharp intensity edges in the background, such as the shadow boundaries, are estimated as occlusion boundaries and the recovered depths around those edges are inaccurate. In comparison, our method estimate the occlusion matting and depths pixelwisely , so it can handle sharp edges in the background and high spatial frequency depth changes for thin structures like mesh, grass and bush branches.

We apply our method to in-vivo micro-scale images of capillaries with diameter less than 50 μm. We use the Braedius CytoCam Camera to capture focal stacks of micro-vessels on the tongue of pigs. The focal planes distance range from 20 μm to 240 μm with step size of 20μm. As shown in Figure 3.7, the occlusion matting and depths of micro-vessels are estimated in the presence of

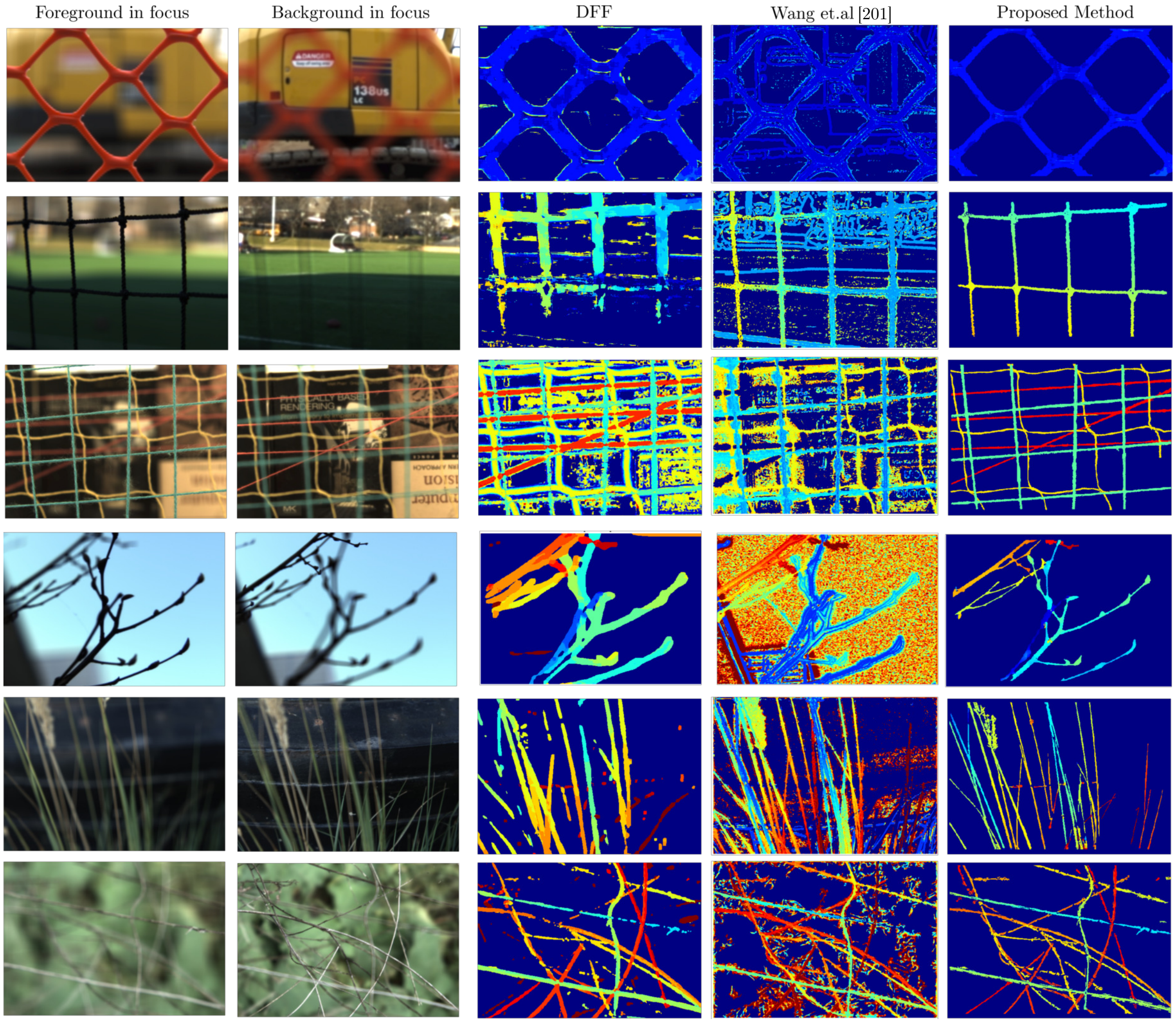


Figure 3.5: The depth recovery for the thin structures. Note that the depth estimations using the DFF method for points close to the occlusion boundaries are inaccurate due to high frequency depth discontinuity. The light field method in [201] does not perform well on the textureless regions and sharp edges in the background. Our method recovers the sharp depth discontinuity on the boundaries of the thin structures such as the grass and bush in the presence of spatially varying defocus blur.

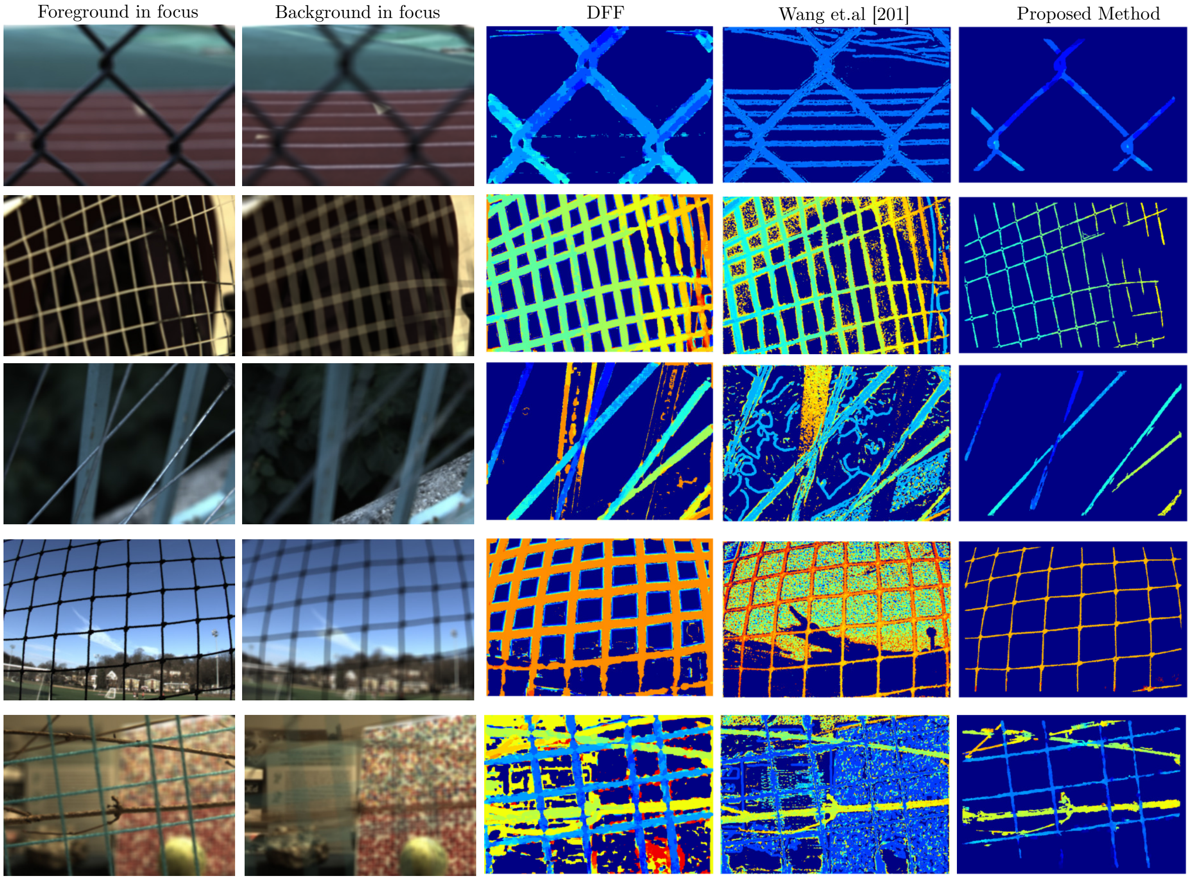
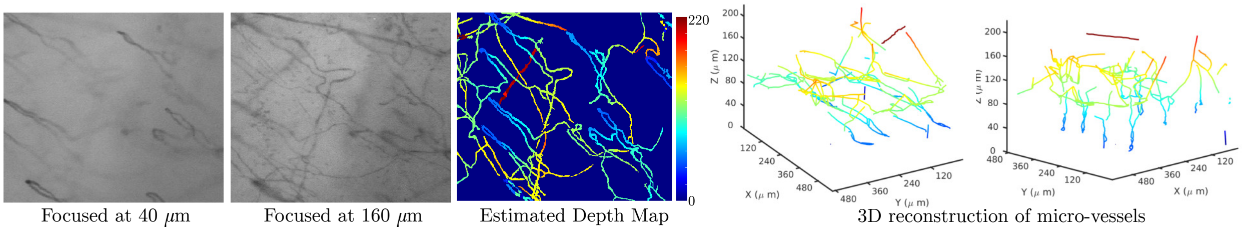


Figure 3.6: The depth recovery for the thin structures. Note that the depth estimations using the DFF method for points close to the occlusion boundaries are inaccurate due to high frequency depth discontinuity. The light field method in [201] does not perform well on the textureless regions and sharp edges in the background. Our method recovers the sharp depth discontinuity on the boundaries of the thin structures.



Focused at 40 μm

Focused at 160 μm

Estimated Depth Map

3D reconstruction of micro-vessels

Figure 3.7: The depth map and 3D reconstruction of micro-vessels. From left to right: 2 of 12 images in the focal stack; the estimated depth map, and two views of the reconstructed 3D structure. The 3D reconstruction is color coded to visualize the depth variations. To our knowledge, our method is the first approach to reconstruct the 3D structures of micro-vessels using non-invasive in-vivo image measurements.

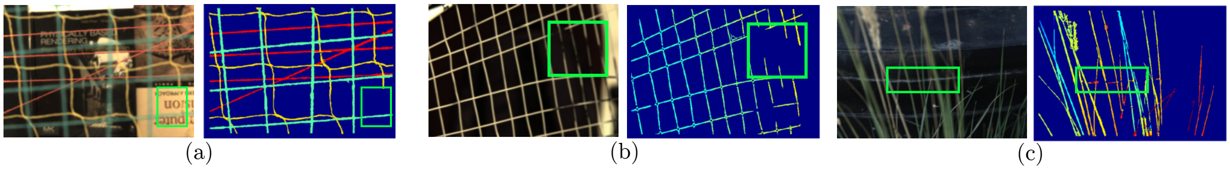


Figure 3.8: Limitations of our method. (a)(b) In the box marked in green, the thin structures in the near field have similar color as the background. In this case, the matting estimation fails because switching the occlusion matte values for the points in those regions will not introduce enough image intensity changes. (c) The depth of the background is close to the depth of the thin occluder, thus the estimated occlusion matte includes edges in the background.

spatially varying defocus blur and occlusions. Then we reconstruct the 3D structure of the micro-vessels based on the depth map. To our knowledge, our method is the first approach to reconstruct the 3D structures of capillaries using non-invasive image measurements.

3.7 Limitation

Our methods have several limitations: First, if the thin structures in the near field have similar color as the background. the occlusion matting estimation may fail because switching the occlusion matte values for the points in those regions will not introduce enough image intensity changes, as shown in Figure 3.8(a)(b); Second, when the depth of the background is close to the depth of the thin occluder, the occlusion matte estimation tends to include the edges in the background as thin structures/occlusion boundaries, as shown in Figure 3.8(c).

3.8 Conclusions

We presented a method for matting and depth recovery for thin structures from a focal stack. We proposed a general image formation model with the spatially varying blur and mutual occlusions explicitly accounted for. Based on the model, for matting, we design an efficient MCMC inference method where the image/model update is computed analytically without explicitly rendering new images. The depth of thin structures is then recovered using gradient descent with the differential terms calculated from the image formation model. We evaluated the proposed method on images of scenes at both macro and micro scales.

We assume that the sizes/widths of objects are small compared to the aperture. In addition, if the foreground objects are far away from the camera, the camera model degrades to a pinhole camera model and the image formation model in Section 3.3 is invalid. To handle larger/distant occlusions, we can extend the method to include multiple cameras such that a large synthetic aperture [194] can be obtained. Another future direction is to extend the approach to scenes with transparent or semi-transparent occlusions, such as smoke, glass, and water droplets.

Chapter 4

Depth and Uncertainty from a Video Camera

4.1 Introduction

Depth sensing is crucial for 3D reconstruction [134, 137, 204] and scene understanding [74, 150, 173]. Active depth sensors (*e.g.*, time of flight cameras [82, 153], LiDAR [34]) measure dense metric depth, but often have limited operating range (*e.g.*, indoor) and spatial resolution [23], consume more power, and suffer from multi-path reflection and interference between sensors [120]. In contrast, estimating depth directly from image(s) solves these issues, but faces other long-standing challenges such as scale ambiguity and drift for monocular methods [158], as well as the correspondence problem and high computational cost for stereo [186] and multi-view methods [164].

Inspired by recent success of deep learning in 3D vision [15, 28, 60, 67, 85, 182, 193, 199, 214, 220, 221], in this chapter, we propose a DL-based method to estimate depth and its uncertainty continuously from a monocular video stream, with the goal of effectively turning an RGB camera into an RGB-D camera. We have two key ideas:

1. Unlike prior work, for each pixel, we estimate a depth probability distribution rather than a single depth value, leading to an estimate of a Depth Probability Volume (DPV) for each input frame. The DPV provides both a Maximum-Likelihood-Estimate (MLE) of the depth map, as well as the corresponding per-pixel uncertainty measure.
2. These DPVs across different frames are accumulated over time, as more incoming frames are processed sequentially. The accumulation step, originated from the Bayesian filtering theory and implemented as a learnable deep network, effectively reduces depth uncertainty and improves accuracy, robustness, and temporal stability over time, as shown later in Sec. 4.4.

We argue that all DL-based depth estimation methods should predict *not depth values but depth distributions*, and should *integrate such statistical distributions over time* (*e.g.*, via Bayesian filtering). This is because dense depth estimation from image(s) – especially for single-view methods – inherently has a lot of uncertainty, due to factors such as lack of texture, specular/transparent material, occlusion, and scale drift. While some recent work started focusing on uncertainty estimation [63, 86, 95, 96] for certain computer vision tasks, to our knowledge, we are the first to predict a depth probability volume from images and integrate it over time in a statistical framework.

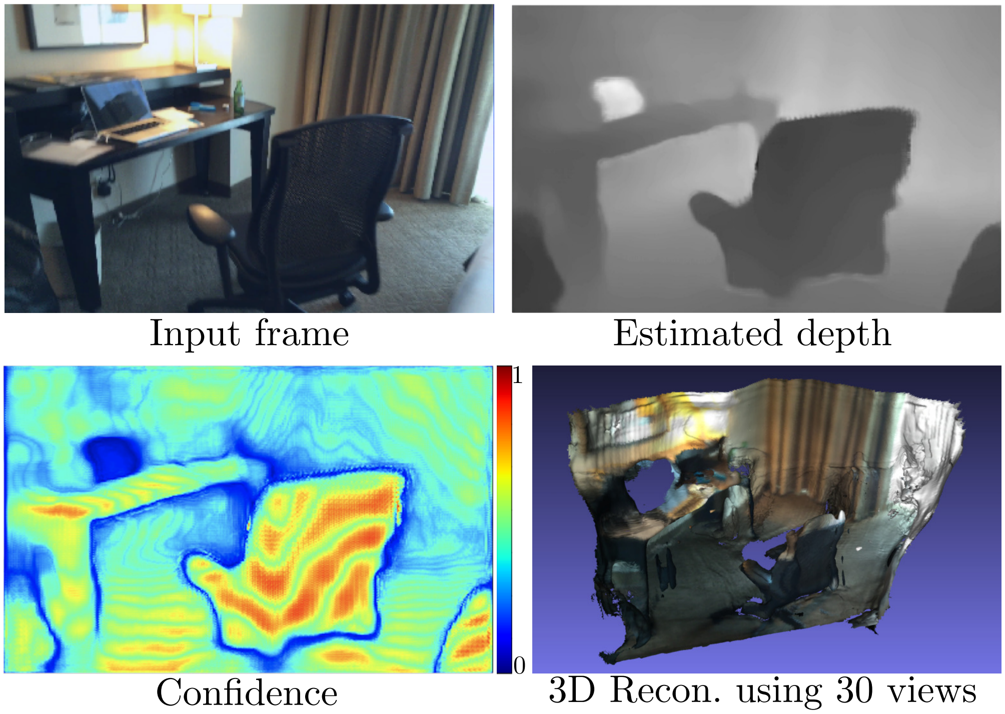


Figure 4.1: We proposed a DL-based method to estimate depth and its uncertainty (or, confidence) continuously for a monocular video stream, with the goal of turning an RGB camera into an RGB-D camera. Its output can be directly fed into classical RGB-D based 3D scanning methods [134, 137] for 3D reconstruction.

We evaluate our method extensively on multiple datasets and compare with recent state-of-the-art, DL-based, depth estimation methods [60, 67, 193]. We also perform the so-called "cross-dataset" evaluation task, which tests models trained on a different dataset without fine-tuning. We believe such cross-dataset tasks are essential to evaluate the robustness and generalization ability [3]. Experimental results show that, with reasonably good camera pose estimation, our method outperforms these prior methods on depth estimation with better accuracy, robustness, and temporal stability. Moreover, the output of the proposed method can be directly fed into RGB-D based 3D scanning methods [134, 137] for 3D scene reconstruction.

4.2 Related Work

Depth sensing from active sensors Active depth sensors, such as depth cameras [82, 153] or LiDAR sensors [34] provide dense metric depth measurements as well as sensor-specific confidence measure [154]. Despite of their wide usage [74, 134, 150, 204], they have several inherent drawbacks[23, 120, 147, 191], such as limited operating range, low spatial resolution, sensor interference, and high power consumption. Our goal in this chapter is to mimic a RGB-D sensor with a monocular RGB camera, which continuously predicts depth (and its uncertainty) from a video stream.

Depth estimation from images Depth estimation directly from images has been a core problem in computer vision [159, 164]. Classical single view methods [39, 158] often make strong assumptions on scene structures. Stereo and multi-view methods [164] rely on triangulation and suffer from finding correspondences for textureless regions, transparent/specular materials, and occlusion. Moreover, due to global bundle adjustment, these methods are often computationally expensive for real-time applications. For depth estimation from a monocular video, there is also scale ambiguity and drifting [128]. Because of these challenges, many computer vision systems [128, 161] use RGB images mainly for camera pose estimation but rarely for dense 3D reconstruction [162]. Nevertheless, depth sensing from images has great potentials, since it addresses all the above drawbacks of active depth sensors. In this chapter, we take a step in this direction using a learning-based method.

Learning-based depth estimation Recently researchers have shown encouraging results for depth sensing directly from images(s), including single-view methods [60, 67, 221], video-based methods [119, 199, 216], depth and motion from two views [28, 193], and multi-view stereo [85, 214, 220]. A few work also incorporated these DL-based depth sensing methods into visual SLAM systems [15, 182]. Despite of the promising performance, however, these DL-based methods are still far from real-world applications, since their robustness and generalization ability is yet to be thoroughly tested [3]. In fact, as shown in Sec. 4.4, we found many state-of-the-art methods degrade significantly even for simple cross-dataset tasks. This gives rise to an increasing demand for a systematic study of uncertainty and Bayesian deep learning for depth sensing, as performed in this chapter.

Uncertainty and Bayesian deep learning Uncertainty and Bayesian modeling have been long studied in last few decades, with various definitions ranging from the variance of posterior distributions for low-level vision [179] and motion analysis [97] to variability of sensor input models [94]. Recently, uncertainty [63, 95] for Bayesian deep learning were introduced for a variety of computer vision tasks [36, 86, 96]. In our work, the uncertainty is defined as the posterior probability of depth, *i.e.*, the DPV estimated from a local window of several consecutive frames. Thus, our network estimates the “measurement uncertainty” [95] rather than the “model uncertainty”. We also learn an additional network module to integrate this depth probability distribution over time in a Bayesian filtering manner, in order to improve the accuracy and robustness for depth estimation from a video stream.

4.3 Our Approach

Figure 4.2 shows an overview of our proposed method for depth sensing from an input video stream, which consists of three parts. The first part (Sec. 4.3.1) is the D-Net, which estimates the Depth Probability Volume (DPV) for each input frame. The second part (Sec. 4.3.2) is the K-Net,

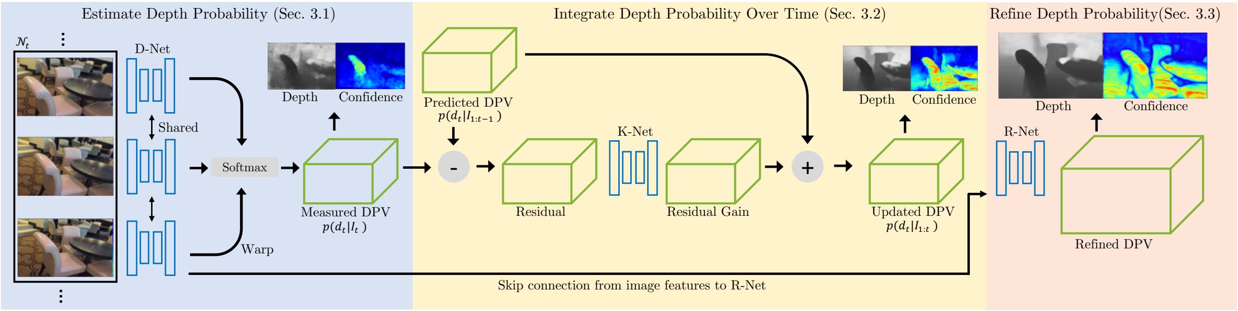


Figure 4.2: Overview of the proposed network for depth estimation with uncertainty from a video. Our method takes the frames in a local time window in the video as input and outputs a Depth Probability Volume (DPV) that is updated over time. The update procedure is in a Bayesian filter fashion: we first take the difference between the local DPV estimated using the D-Net (Sec. 4.3.1) and the predicted DPV from previous frames to get the residual; then the residual is modified by the K-Net (Sec. 4.3.2) and added back to the predicted DPV; at last the DPV is refined and upsampled by the R-Net (Sec. 4.3.3), which can be used to compute the depth map and its confidence measure.

which helps to integrate the DPVs over time. The third part (Sec. 4.3.3) is the refinement R-Net, which improves the spatial resolution of DPVs with the guidance from input images.

Specifically, we denote the depth probability volume (DPV) as $p(d; u, v)$, which represents the probability of pixel (u, v) having a depth value d, where $d \in [d_{min}, d_{max}]$. Due to perspective projection, the DPV is defined on the 3D view frustum attached to the camera, as shown in Fig. 4.3(a). d_{min} and d_{max} are the near and far planes of the 3D frustum, which is discretized into $N = 64$ planes uniformly over the inverse of depth (*i.e.*, disparity). The DPV contains the complete statistical distribution of depth for a given scene. In this chapter, we directly use the non-parametric volume to represent DPV. Parametric models, such as Gaussian Mixture Model [14], can be also be used. Given the DPV, we can compute the Maximum-Likelihood Estimates (MLE) for depth and its confidence:

$$\text{Depth}: \quad \hat{d}(u, v) = \sum_{d=d_{min}}^{d=d_{max}} p(d; (u, v)) \cdot d, \tag{4.1}$$

$$\text{Confidence}: \quad \hat{C}(u, v) = p(\hat{d}, (u, v)). \tag{4.2}$$

To make notations more concise, we will omit (u, v) and use $p(d)$ for DPVs in the rest of the chapter.

When processing a video stream, the DPV can be treated as a hidden state of the system. As the camera moves, as shown in Fig. 4.3(b), the DPV $p(d)$ is being *updated* as new observations arrive, especially for the overlapping volumes. Meanwhile, if camera motion is known, we can easily *predict* the next state $p(d)$ from the current state. This predict-update iteration naturally implies a Bayesian filtering scheme to update the DPV over time for better accuracy.

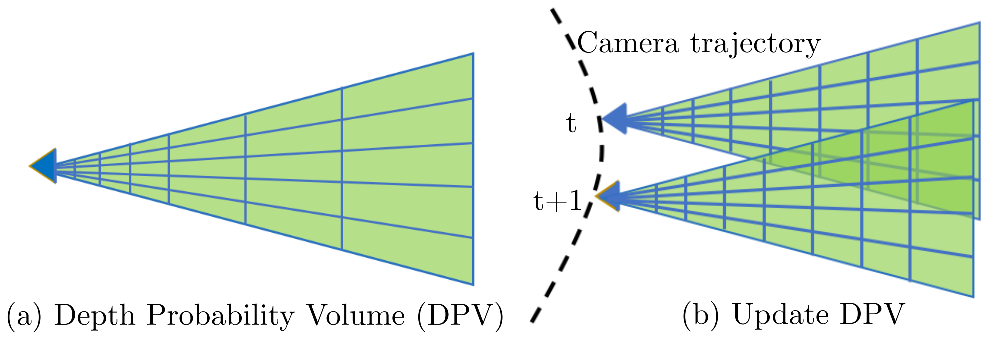


Figure 4.3: Representation and update for DPV. (a) The DPV is defined over a 3D frustrum defined by the pinhole camera model. (b) The DPV gets updated over time as the camera moves.

4.3.1 D-Net: Estimating DPV

For each frame I_t, we use a CNN, named D-Net, to estimate the conditional DPV, $p(d_t|I_t)$, using I_t and its temporally neighboring frames. In this chapter, we consider a local time window of five frames $\mathcal{N}_t = [t - 2\Delta t, t - \Delta t, t, t + \Delta t, t + 2\Delta t]$, and we set $\Delta t = 5$ for all our testing videos (25fps/30fps). For a given depth candidate d, we can compute a cost map by warping all the neighboring frames into the current frame I_t and computing their differences. Thus, for all depth candidates, we can compute a cost volume, which produces the DPV after a softmax layer:

$$\begin{aligned} L(d_t|I_t) &= \sum_{k \in \mathcal{N}_t, k \neq t} ||f(I_t) - \text{warp}(f(I_k); d_t, \delta T_{kt})||, \\ p(d_t|I_t) &= \text{softmax}(L(d_t|I_t)), \end{aligned} \tag{4.3}$$

where $f(\cdot)$ is a feature extractor, δT_{kt} is the relative camera pose from frame I_k to frame I_t, warp$(\cdot)$ is an operator that warps the image features from frame I_k to the reference frame I_t, which is implemented as 2D grid sampling. In this chapter, without loss of generality, we use the feature extractor $f(\cdot)$ from PSM-Net [28], which outputs a feature map of 1/4 size of the input image. Later in Sec. 4.3.3, we learn a refinement R-Net to upsample the DPV back to the original size of the input image.

Figure 4.4 shows an example of a depth map $\hat{d}(u, v)$ and its confidence map $\hat{C}(u, v)$ (blue means low confidence) derived from a Depth Probability Volume (DPV) from the input image. The bottom plot shows the depth probability distributions $p(d; u, v)$ for the three selected points, respectively. The red and green points have sharp peaks, which indicates high confidence in their depth values. The blue point is in the highlight region, and thus it has a flat depth probability distribution and a low confidence for its depth.

4.3.2 K-Net: Integrating DPV over Time

When processing a video stream, our goal is to integrate the local estimation of DPVs over time to reduce uncertainty. As mentioned earlier, this integration can be naturally implemented as

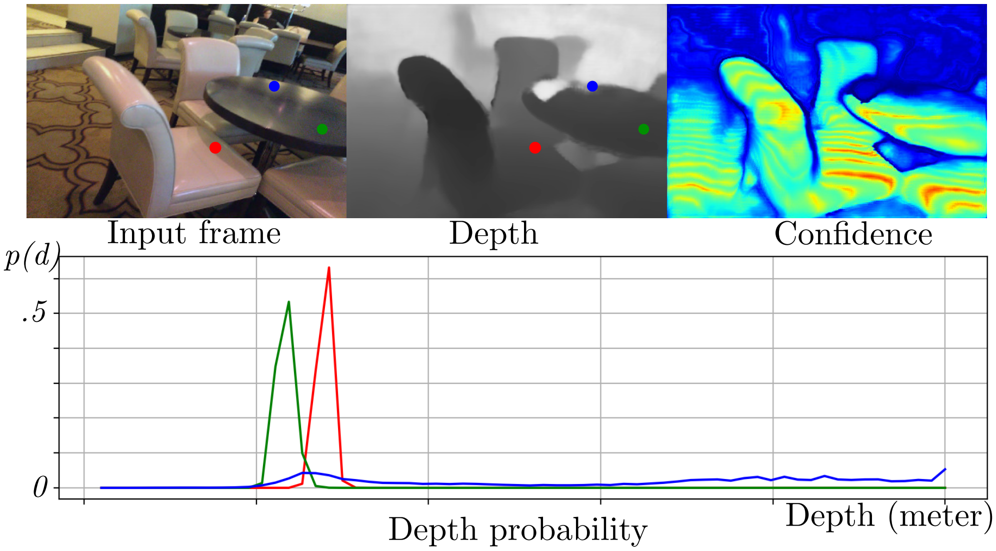


Figure 4.4: An example of a depth map $\hat{d}(u, v)$ and its confidence map $\hat{C}(u, v)$ (blue means low confidence) derived from a Depth Probability Volume (DPV). The bottom plot shows the depth probability distributions $p(d; u, v)$ for the three selected points, respectively. The red and green points have sharp peaks, which indicates high confidence in their depth values. The blue point is in the highlight region, which results in a flat depth probability distribution and a low confidence for its depth value.

Bayesian filtering. Let us define d_t as the hidden state, which is the depth (in camera coordinates) at frame I_t. The "belief" volume $p(d_t|I_{1:t})$ is the conditional distribution of the state giving all the previous frames. A simple Bayesian filtering can be implemented in two iterative steps:

$$\begin{aligned} \text{Predict}: &\quad p(d_t|I_{1:t}) \to p(d_{t+1}|I_{1:t}), \\ \text{Update}: &\quad p(d_{t+1}|I_{1:t}) \to p(d_{t+1}|I_{1:t+1}), \end{aligned} \tag{4.4}$$

where the prediction step is to warp the current DPV from the camera coordinate at t to the camera coordinate at $t + 1$:

$$p(d_{t+1}|I_{1:t}) = \text{warp}(p(d_t|I_{1:t}), \delta T_{t,t+1}), \tag{4.5}$$

where $\delta T_{t,t+1}$ is the relative camera pose from time t to time $t + 1$, and warp$(\cdot)$ here is a warping operator implemented as 3D grid sampling. At time $t + 1$, we can compute the local DPV $p(d_{t+1}|I_{t+1})$ from the new measurement I_{t+1} using the D-Net. This local estimate is thus used to update the hidden state, *i.e.*, the "belief" volume,

$$p(d_{t+1}|I_{1:t+1}) = p(d_{t+1}|I_{1:t}) \cdot p(d_{t+1}|I_{t+1}). \tag{4.6}$$

Note that we always normalize the DPV in the above equations and ensure $\int_{d_{min}}^{d_{max}} p(d) = 1$. Figure 4.5 shows an example. As shown in the second row, with the above Bayesian filtering (labeled as ”no damping”), the estimated depth map is less noisy, especially in the regions of the back wall and the floor.

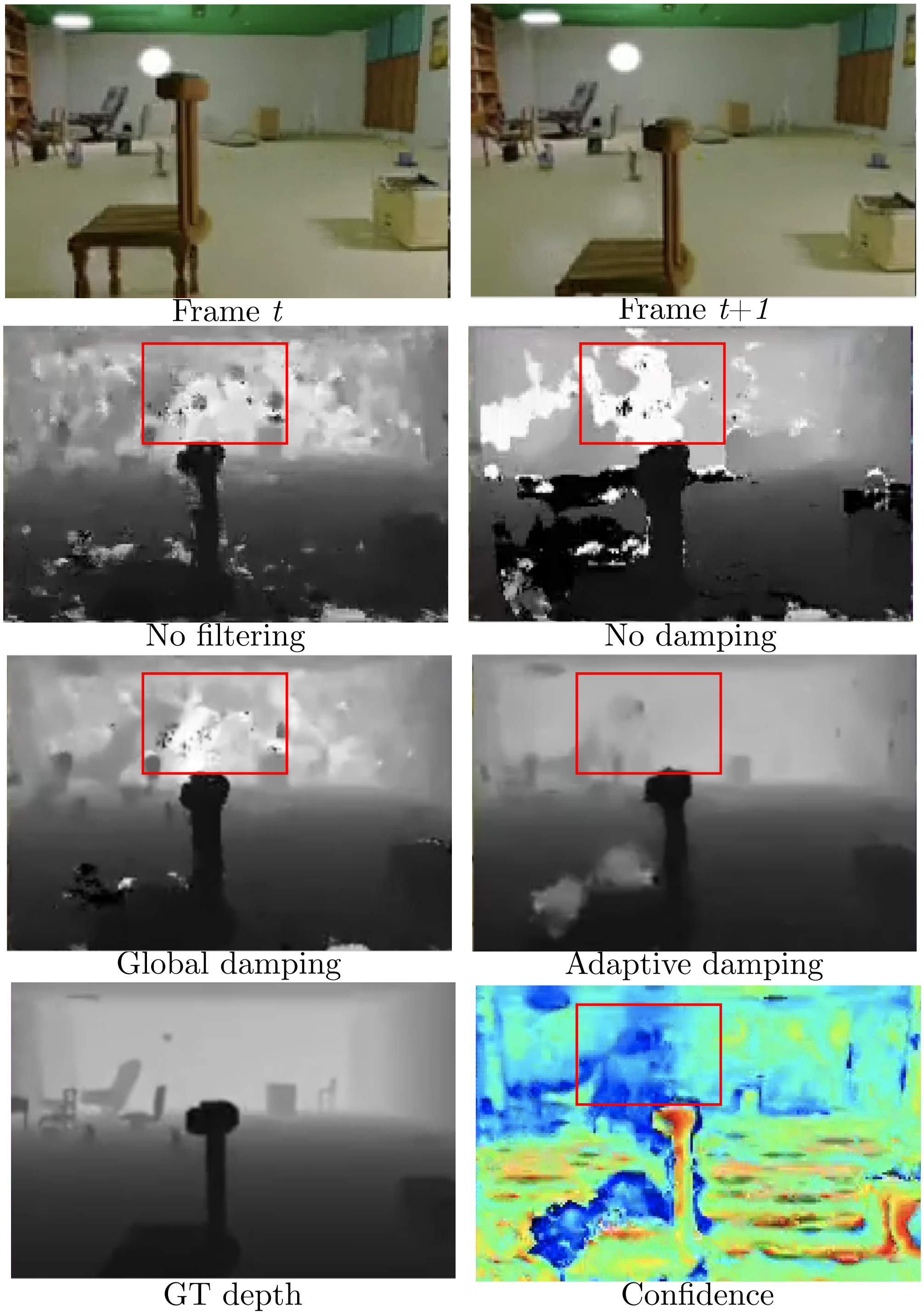


Figure 4.5: Comparison between different methods for integrating DPV over time. Part of the wall is occluded by the chair at frame t and disoccluded in frame $t+1$. **No filtering**: not integrating the DPV over time. **No damping**: integrating DPV directly with Bayesian filtering. **Global damping**: down-weighting the predicted DPV for all voxels using Eq. 4.7 with $\lambda = 0.8$. **Adaptive damping**: down-weighting the predicted DPV adaptively with the K-Net (Sec. 4.3.2). Using the K-net, we get the best depth estimation for regions with/without disocclusion.

However, one problem with directly applying Bayesian filtering is it integrates both correct and incorrect information in the prediction step. For example, when there are occlusions or disocclusions, the depth values near the occlusion boundaries change abruptly. Applying Bayesian filtering directly will propagate wrong information to the next frames for those regions, as highlighted in the red box in Fig. 4.5. One straightforward solution is to reduce the weight of the prediction in order to prevent incorrect information being integrated over time. Specifically, by defining $E(d) = -\log p(d)$, Eq. 4.6 can be re-written as

$$E(d_{t+1}|I_{1:t+1}) = E(d_{t+1}|I_{1:t}) + E(d_{t+1}|I_{t+1}),$$

where the first term is the prediction and the second term is the measurement. To reduce the weight of the prediction, we multiply a weight $\lambda \in [0, 1]$ with the first term,

$$E(d_{t+1}|I_{1:t+1}) = \lambda \cdot E(d_{t+1}|I_{1:t}) + E(d_{t+1}|I_{t+1}). \tag{4.7}$$

We call this scheme "global damping". As shown in Fig. 4.5, global damping helps to reduce the error in the disoccluded regions. However, global damping may also prevent some correct depth information to be integrated to the next frames, since it reduces the weights equally for all voxels in the DPV. Therefore, we propose an "adaptive damping" scheme to update the DPV:

$$E(d_{t+1}|I_{1:t+1}) = E(d_{t+1}|I_{1:t}) + g(\Delta E_{t+1}, I_{t+1}), \tag{4.8}$$

where ΔE_{t+1} is the difference between the measurement and the prediction,

$$\Delta E_{t+1} = E(d_{t+1}|I_{t+1}) - E(d_{t+1}|I_{1:t}), \tag{4.9}$$

and $g(\cdot)$ is a CNN, named K-Net, which learns to transform ΔE_{t+1} into a correction term to the prediction. Intuitively, for regions with correct depth probability estimates, the values in the overlapping volume of DPVs are consistent. Thus the residual in Eq. 4.9 is small and the DPV will not be updated in Eq. 4.8. On the other hand, for regions with incorrect depth probability, the residual would be large and the DPV will be corrected by $g(\Delta E, I_{t+1})$. This way, the weight for prediction will be changed adaptively for different DPV voxels. As shown in Fig. 4.5, the adaptive damping, *i.e.*, K-Net, significantly improve the accuracy for depth estimation.

4.3.3 R-Net and Training Details

Finally, since the DPV $p(d_t|I_{1:t})$ is estimated with $1/4$ spatial resolution (on both width and height) of the input image, we employ a CNN, named R-Net, to upsample and refine the DPV back to the original image resolution. The R-Net, $h(\cdot)$, is essentially an U-Net with skip connections, which takes input the low-res DPV from the K-Net $g(\cdot)$ and the image features extracted from the feature extractor $f(\cdot)$, and outputs a high-resolution DPV.

In summary, as shown in Fig. 4.2, the entire network has three modules, *i.e.*, the D-Net, $f(\cdot; \Theta_1)$, the K-Net, $g(\cdot; \Theta_2)$, and the R-Net, $h(\cdot; \Theta_3)$. Detailed network architectures are provided in Fig. 4.6, Fig. 4.7 and Fig. 4.8. The full network is trained end-to-end, with simply the Negative Log-Likelihood (NLL) loss over the depth, Loss $=$ NLL$(p(d), d_{GT})$. We also tried to add image warping as an additional loss term (*i.e.*, minimizing the difference between I_t and the warped neighboring frames), but we found that it does not improve the quality of depth prediction.

During training, we use ground truth camera poses. For all our experiments, we use the ADAM optimizer [98] with a learning rate of 10^{-5}, $\beta_1 = .9$ and $\beta_2 = .999$. The whole framework, including D-Net, K-Net and R-Net, is trained together in an end-to-end fashion for 20 epochs.

4.3.4 Camera Poses during Inference

During inference, given an input video stream, our method requires relative camera poses δT between consecutive frames — at least for all the first five frames — to bootstrap the computation of the DPV. In this chapter, we evaluated several options to solve this problem. In many applications, such as autonomous driving and AR, initial camera poses may be provided by additional sensors such as GPS, odometer, or IMU. Alternatively, we can also run state-of-the-art monocular visual odometry methods, such as DSO [53], to obtain the initial camera poses. Since our method outputs continuous dense depth maps and their uncertainty maps, we can in fact further optimize the initial camera poses within a local time window, similar to local bundle adjustment [190].

Specifically, as shown in Fig. 4.9, given $p(d_t|I_{1:t})$, the DPV of the reference frame I_t in the local time window $\mathcal{N}_t$, we can warp $p(d_t|I_{1:t})$ to the reference camera view in $\mathcal{N}_{t+1}$ to predict the DPV $p(d_{t+1}|I_{1:t})$ using Eq. 4.5. Then we get the depth map $\hat{d}$ and confidence map $\hat{C}$ for the new reference frame using Eq. 4.2. The camera poses within the local time window $\mathcal{N}_{t+1}$ are optimized as:

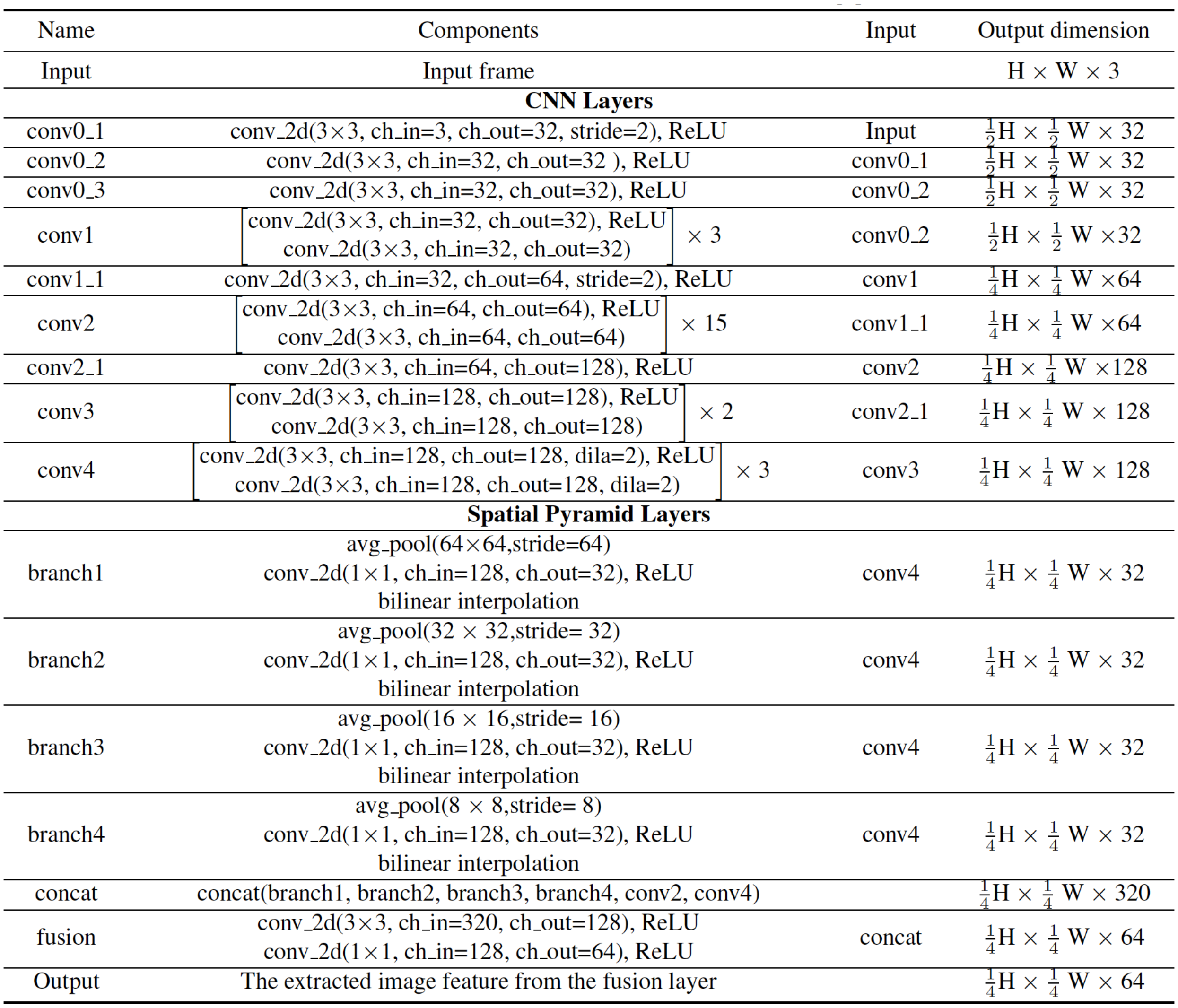
$$\min_{\substack{\delta T_{k,t+1} \\ k \in \mathcal{N}_{t+1}, k \neq t+1}} \sum_k \hat{C} |I_{t+1} - \text{warp}(I_k; \hat{d}; \delta T_{k,t+1})|_1, \tag{4.10}$$

where $\delta T_{k,t+1}$ is the relative camera pose of frame k to frame $t+1$; I_k is the source image at frame k; warp$(\cdot)$ is an operator that warps the image from the source view to the reference view.

4.4 Experimental Results

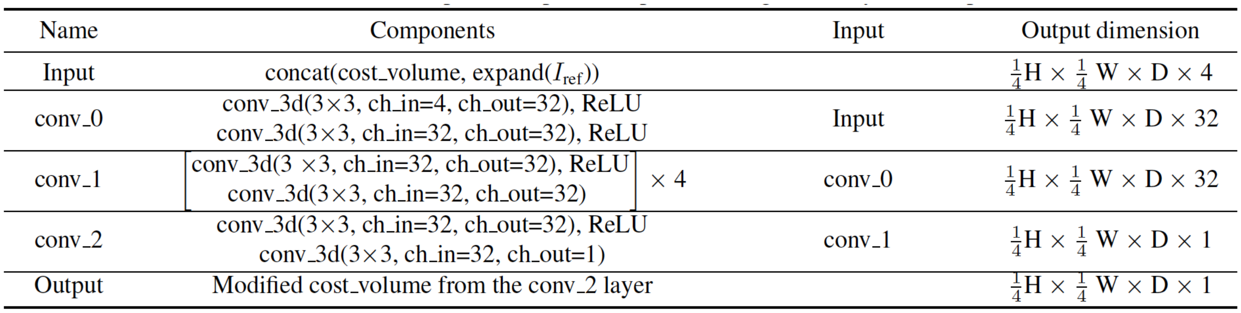
We evaluate our method on multiple indoor and outdoor datasets [62, 64, 168, 176], with an emphasis on accuracy and robustness. For accuracy evaluation, we argue the widely-used statistical metrics [52, 193] are insufficient because they can only provide an overall estimate over the entire depth map. Rather, we feed the estimated depth maps directly into classical RGB-D based 3D scanning systems [134, 137] for 3D reconstruction — this will show the metric accuracy, the consistency, and the usefulness of the estimation. For robustness evaluation, we performed the aforementioned cross-dataset evaluation tasks, *i.e.*, testing on new datasets without fine-tuning. The performance degradation over new datasets will show the generalization ability and robustness for a given algorithm.

As no prior work operates in the exact setting as ours, it is difficult to choose methods to compare with. We carefully select a few recent DL-based depth estimation methods and try our best for a fair comparison. For single-view methods, we select DORN [60] which is the current state-of-the-art [3]. For two-view methods, we compare with DeMoN [193], which shows high quality depth prediction from a pair of images. We also compare with MonoDepth [67], which is



Name	Components	Input	Output dimension
Input	Input frame		H × W × 3
	CNN Layers		
conv0_1	conv_2d(3×3, ch_in=3, ch_out=32, stride=2), ReLU	Input	$\frac{1}{2}$H × $\frac{1}{2}$ W × 32
conv0_2	conv_2d(3×3, ch_in=32, ch_out=32), ReLU	conv0_1	$\frac{1}{2}$H × $\frac{1}{2}$ W × 32
conv0_3	conv_2d(3×3, ch_in=32, ch_out=32), ReLU	conv0_2	$\frac{1}{2}$H × $\frac{1}{2}$ W × 32
conv1	[conv_2d(3×3, ch_in=32, ch_out=32), ReLU conv_2d(3×3, ch_in=32, ch_out=32)] × 3	conv0_2	$\frac{1}{2}$H × $\frac{1}{2}$ W ×32
conv1_1	conv_2d(3×3, ch_in=32, ch_out=64, stride=2), ReLU	conv1	$\frac{1}{4}$H × $\frac{1}{4}$ W ×64
conv2	[conv_2d(3×3, ch_in=64, ch_out=64), ReLU conv_2d(3×3, ch_in=64, ch_out=64)] × 15	conv1_1	$\frac{1}{4}$H × $\frac{1}{4}$ W ×64
conv2_1	conv_2d(3×3, ch_in=64, ch_out=128), ReLU	conv2	$\frac{1}{4}$H × $\frac{1}{4}$ W ×128
conv3	[conv_2d(3×3, ch_in=128, ch_out=128), ReLU conv_2d(3×3, ch_in=128, ch_out=128)] × 2	conv2_1	$\frac{1}{4}$H × $\frac{1}{4}$ W × 128
conv4	[conv_2d(3×3, ch_in=128, ch_out=128, dila=2), ReLU conv_2d(3×3, ch_in=128, ch_out=128, dila=2)] × 3	conv3	$\frac{1}{4}$H × $\frac{1}{4}$ W × 128
	Spatial Pyramid Layers		
branch1	avg_pool(64×64,stride=64) conv_2d(1×1, ch_in=128, ch_out=32), ReLU bilinear interpolation	conv4	$\frac{1}{4}$H × $\frac{1}{4}$ W × 32
branch2	avg_pool(32 × 32,stride= 32) conv_2d(1×1, ch_in=128, ch_out=32), ReLU bilinear interpolation	conv4	$\frac{1}{4}$H × $\frac{1}{4}$ W × 32
branch3	avg_pool(16 × 16,stride= 16) conv_2d(1×1, ch_in=128, ch_out=32), ReLU bilinear interpolation	conv4	$\frac{1}{4}$H × $\frac{1}{4}$ W × 32
branch4	avg_pool(8 × 8,stride= 8) conv_2d(1×1, ch_in=128, ch_out=32), ReLU bilinear interpolation	conv4	$\frac{1}{4}$H × $\frac{1}{4}$ W × 32
concat	concat(branch1, branch2, branch3, branch4, conv2, conv4)		$\frac{1}{4}$H × $\frac{1}{4}$ W × 320
fusion	conv_2d(3×3, ch_in=320, ch_out=128), ReLU conv_2d(1×1, ch_in=128, ch_out=64), ReLU	concat	$\frac{1}{4}$H × $\frac{1}{4}$ W × 64
Output	The extracted image feature from the fusion layer		$\frac{1}{4}$H × $\frac{1}{4}$ W × 64

Figure 4.6: D-Net structure. The structure is taken from [28]



Name	Components	Input	Output dimension
Input	concat(cost_volume, expand(I_{ref}))		$\frac{1}{4}$H × $\frac{1}{4}$ W × D × 4
conv_0	conv_3d(3×3, ch_in=4, ch_out=32), ReLU conv_3d(3×3, ch_in=32, ch_out=32), ReLU	Input	$\frac{1}{4}$H × $\frac{1}{4}$ W × D × 32
conv_1	[conv_3d(3 ×3, ch_in=32, ch_out=32), ReLU conv_3d(3×3, ch_in=32, ch_out=32)] × 4	conv_0	$\frac{1}{4}$H × $\frac{1}{4}$ W × D × 32
conv_2	conv_3d(3×3, ch_in=32, ch_out=32), ReLU conv_3d(3×3, ch_in=32, ch_out=1)	conv_1	$\frac{1}{4}$H × $\frac{1}{4}$ W × D × 1
Output	Modified cost_volume from the conv_2 layer		$\frac{1}{4}$H × $\frac{1}{4}$ W × D × 1

Figure 4.7: K-Net structure.

Name	Components	Input	Output dimension
Input	cost_volume from K-Net		$\frac{1}{4}$H × $\frac{1}{4}$ W × D
conv_0	conv_2d(3×3, ch_in=64+D, ch_out= 64+D), LeakyReLU conv_2d(3×3, ch_in=64+D, ch_out= 64+D), LeakyReLU	concat(Input, fusion in D-Net)	$\frac{1}{4}$H × $\frac{1}{4}$ W × (64+D)
trans_conv_0	transpose_conv(4×4, ch_in=64+D, ch_out=D, stride=2), LeakyReLU	conv_0	$\frac{1}{2}$H × $\frac{1}{2}$ W × D
conv_1	conv_2d(3×3, ch_in=32+D, ch_out=32 + D), LeakyReLU conv_2d(3×3, ch_in=32+D, ch_out=32 + D),LeakyReLU	concat(trans_conv_0, conv_1 in D-Net	$\frac{1}{2}$H × $\frac{1}{2}$ W × (D+32)
trans_conv_1	transpose_conv(4×4, ch_in=32+D, ch_out=D, stride=2), LeakyReLU	conv_1	H × W × D
conv_2	conv_2d(3×3, ch_in=3+D, ch_out=3+D), LeakyReLU conv_2d(3×3, ch_in=3+D, ch_out=D), LeakyReLU conv_2d(3×3, ch_in= D, ch_out=D)	concat(trans_conv_1, I_{ref})	H × W × D
Output	Upsampled and refined cost_volume		H × W × D

Figure 4.8: R-Net structure.

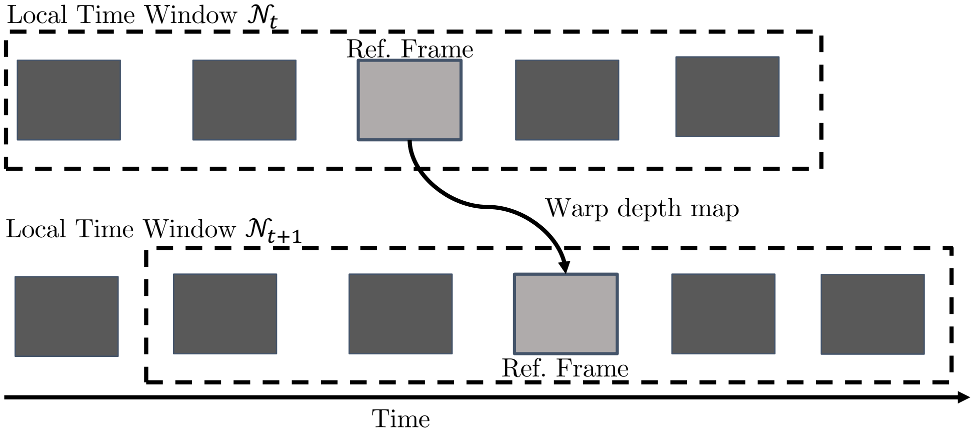

Local Time Window $\mathcal{N}_t$
Ref. Frame
Warp depth map
Local Time Window $\mathcal{N}_{t+1}$
Ref. Frame
Time

Figure 4.9: Camera pose optimization in a sliding local time window during inference. Given the relative camera pose from the reference frame in $\mathcal{N}_t$ to the reference frame in $\mathcal{N}_{t+1}$, we can predict the depth map for the reference frame in $\mathcal{N}_{t+1}$. Then, we optimize the relative camera poses between every source frame and the reference frame in $\mathcal{N}_{t+1}$ using Eq.4.10.

a semi-supervised learning approach from stereo images. To improve the temporal consistency for these per-frame estimations, we trained a post-processing network [102], but we observed it does not improve the performance. Since there is always scale ambiguity for depth from a monocular camera, for fair comparison, we normalize the scale for the outputs from all the above methods before we compute statistical metrics [52].

The inference time for processing one frame in our method is ~ 0.7 second per frame without pose optimization and ~ 1.5 second with pose estimation on a workstation with GTX 1080 GPU and 64 GB RAM memory, with the framework implemented in Python. The pose estimation part

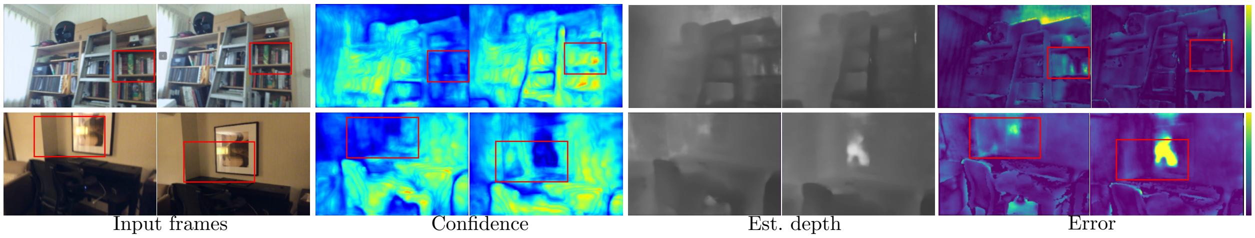


Figure 4.10: Exemplar results of our approach on ScanNet [41]. In addition to high quality depth output, we also obtain reasonable confidence maps (as shown in the marked regions for occlusion and specularity) which correlates with the depth error. Moreover, the confidence maps accumulate correctly over time with more input frames.

Table 4.1: Comparison of depth estimation over the 7-Scenes dataset [168] with the metrics defined in [52].

	$\sigma < 1.25$	abs. rel	rmse	scale inv.
DeMoN [193]	31.88	0.3888	0.8549	0.4473
DORN [60]	60.05	0.2000	0.4591	0.2207
Ours	**69.26**	**0.1758**	**0.4408**	**0.1899**

can be implemented with C++ to improve efficiency.

Results for Indoor Scenarios We first evaluated our method for indoor scenarios, for which RGB-D sensors were used to capture dense metric depth for ground truth. We trained our network on ScanNet [41]. Figure 4.10 shows two exemplar results. As shown, in addition to depth maps, our method also outputs reasonable confidence maps (*e.g.*, low confidence in the occluded or specular regions) which correlates with the depth errors. Moreover, with more input frames, the confidence maps accumulate correctly over time: the confidence of the books (top row) increases and the depth error decreases; the confidence of the glass region (bottom row) decreases and the depth error increases.

For comparison, since the models provided by DORN and DeMoN were trained on different datasets, we compare with these two methods on a separate indoor dataset 7Scenes [168]. For our method, we assume that the relative camera rotation δR within a local time window is provided (*e.g.* measured by IMU). As shown in Table 4.1, our method significantly outperforms both DeMoN and DORN on this dataset based on the commonly used statistical metrics [52]. Without using an IMU, our method can also achieve better performance, as shown in Table 4.4.

For qualitative comparison, as shown in Fig. 4.11, the depth maps from our method are less noisy, more sharper, and temporally more consistent. More importantly, using an RGB-D 3D scanning method [137], we can reconstruct a much higher quality 3D mesh with our estimated depths compared to other methods. Even when compared with 3D reconstruction using a real RGB-D sensor, our result has better coverage and accuracy in some regions (*e.g.*, monitors / glossy

Table 4.2: Comparison of depth estimation on KITTI [64].

	$\sigma < 1.25$	abs. rel	rmse	scale inv.
Eigen [52]	67.80	0.1904	5.114	0.2628
Mono [67]	86.43	0.1238	2.8684	0.1635
DORN [60]	92.62	**0.0874**	3.1375	0.1233
Ours	**93.15**	0.0998	**2.8294**	**0.1070**

Table 4.3: Cross-dataset tests for depth estimation in the outdoors.

	KITTI (train) → virtual KITTI (test)			
	$\sigma < 1.25$	abs. rel	rmse	scale inv.
DORN [60]	69.61	**0.2256**	9.618	0.3986
Ours	**73.38**	0.2537	**6.452**	**0.2548**
	Indoor (train) → KITTI (test)			
	$\sigma < 1.25$	abs. rel	rmse	scale inv.
DORN [60]	25.44	0.6352	8.603	0.4448
Ours	**72.96**	**0.2798**	**5.437**	**0.2139**

surfaces) where active depth sensors cannot capture.

Results for Outdoor Scenarios We also evaluated our method on some outdoor datasets — KITTI [64] and virtual KITTI [62]. The virtual KITTI dataset is used because it has dense, accurate metric depth as ground truth, while KITTI only has sparse depth values from LiDAR as ground truth. For our method, we use the camera poses measured by the IMU and GPS. Table 4.2 lists the comparison results with DORN [60], Eigen [52], and MonoDepth [67] which are also trained on KITTI [64]. Our method has similar performance with DORN [60], and is better than the other two methods, based on the statistical metrics defined in [52].

Figure 4.12 shows qualitative comparison for depth maps in KITTI dataset. As shown, our method generate sharper and less noisier depth maps. In addition, our method outputs depth confidence maps (*e.g.*, lower confidence on the car window). Our depth estimation is temporally consistent, which leads to the possibility of fusing multiple depth maps with voxel hashing [137] in the outdoors for a large-scale dense 3D reconstruction, as shown in Fig. 4.12.

In Table 4.3, we performed the cross-dataset task. The left shows the results with training from KITTI [64] and testing on virtual KITTI [62]. The right shows the results with training from indoor datasets (NYUv2 [131] for DORN [60] and ScanNet [41] for ours) and testing on KITTI [64]. As shown, our method performs better, which shows its better robustness and generalization ability.

Ablation Study The performance of our method relies on accurate estimate of camera poses, so we test our method with different camera pose estimation schemes: (1) Relative camera rotation

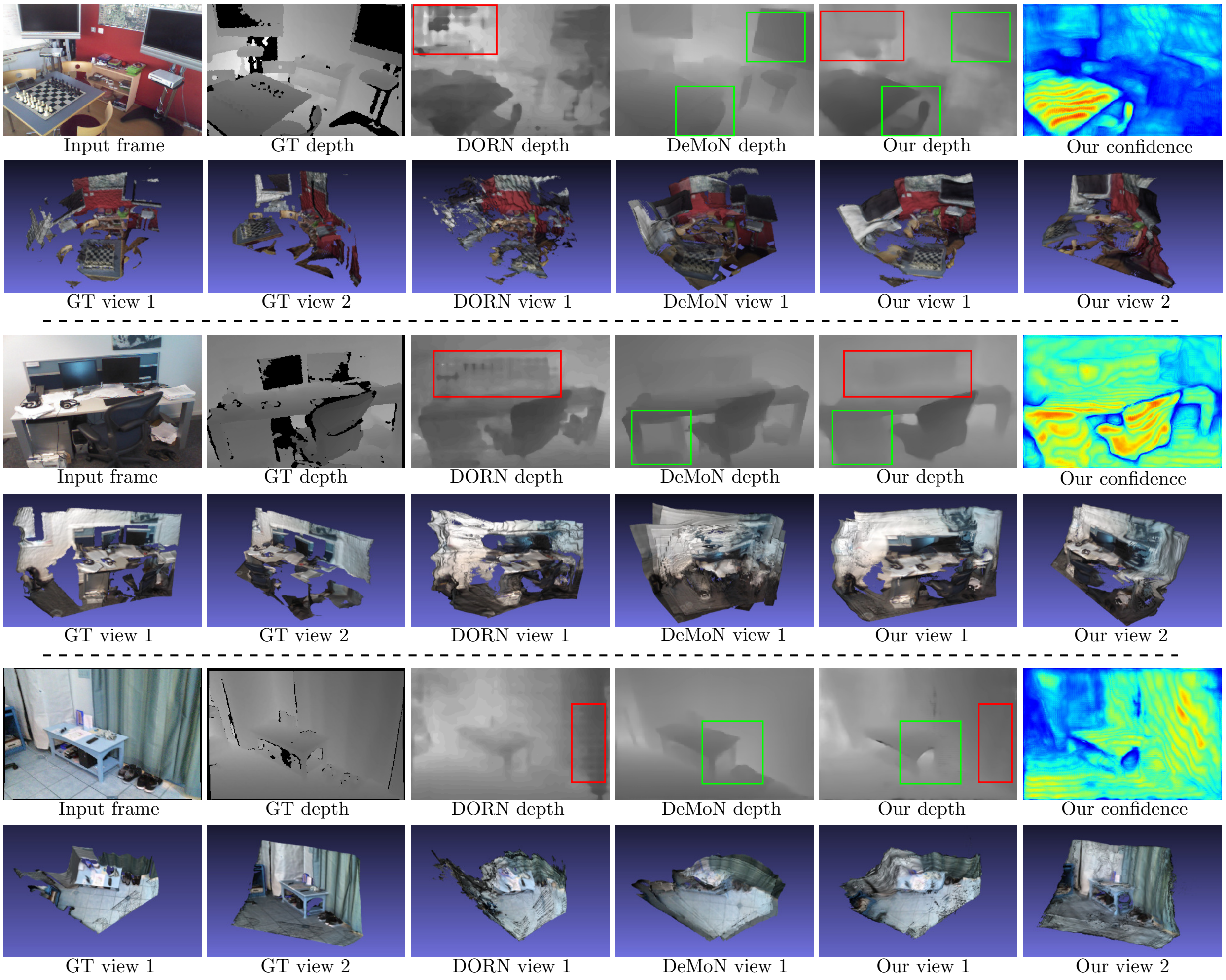
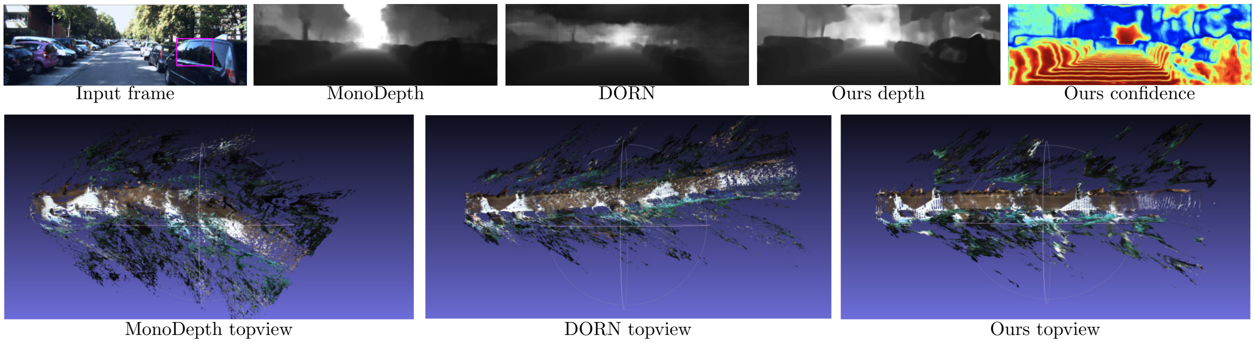


Figure 4.11: Depth and 3D reconstruction results on indoor datasets (best viewed when zoomed in). We compare our method with DORN [60] and DeMoN [193], in terms of both depth maps and 3D reconstruction using Voxel Hashing [137] that accumulates the estimated depth maps for multiple frames. To show the temporal consistency of the depths, we use different numbers of depth maps for Voxel Hashing: 2 depth maps for the first sample and 30 depth maps for the other samples. The depth maps from DORN contain block artifacts as marked in red boxes. This is manifested as the rippled shapes in the 3D reconstruction. DeMoN generates sharp depth boundaries but fails to recover the depth faithfully in the regions marked in the green box. Also, the depths from DeMoN is not temporally consistent. This leads to the severe misalignment artifacts in the 3D reconstructions. In comparison, our method generates correct and temporally consistent depths maps, especially at regions with high confidence, such as the monitor where even the Kinect sensor fails to get the depth due to low reflectance.


Input frame

MonoDepth

DORN

Ours depth

Ours confidence

MonoDepth topview

DORN topview

Ours topview

Figure 4.12: Depth map and 3D reconstruction for KITTI, compared with DORN [60], MonoDepth [193] (best viewed when zoomed in). First row: Our depth map is sharper and contains less noise. For specular region (marked in the pink box), the confidence is lower. Second row, from left to right: reconstructions using depth maps of the same 100 frames estimated from MonoDepth, DORN and our method. All meshes are viewed from above. Within the 100 frames, the vehicle was travelling in a straight line without turning.

Table 4.4: Performance on 7Scenes with different initial poses

	$\sigma < 1.25$	abs. rel	rmse	scale inv.
VO pose	60.63	0.1999	0.4816	0.2158
$1st$ win.	62.08	0.1923	0.4591	0.2001
GT R	69.26	0.1758	0.4408	0.1899
GT pose	70.54	0.1619	0.3932	0.1586

δR is read from an IMU sensor (denoted as "GT R"). (2) δR of all frames are initialized with DSO [53] (denoted as "VO pose") (3) δR of the first five frames are initialized with DSO [53] (denoted as "$1st$ win"). We observe that when only the camera poses in the first time window are initialized using DSO, the performance in terms of depth estimation is better than that using the DSO pose initialization for all frames. This may seem counter-intuitive, but it is because monocular VO methods sometimes have large errors for textureless regions while optimization with dense depths may overcome this problem.

Usefulness of the Confidence Map The estimated confidence maps can also be used to further improve the depth maps. As shown in Fig. 4.13(a), given the depth map and the corresponding confidence, we can correct the regions with lower confidence due to specular reflection. Also, for 3D reconstruction algorithm, given the depth confidence, we can mask out the regions with lower confidence for better reconstruction, as shown in Fig. 4.13(b).

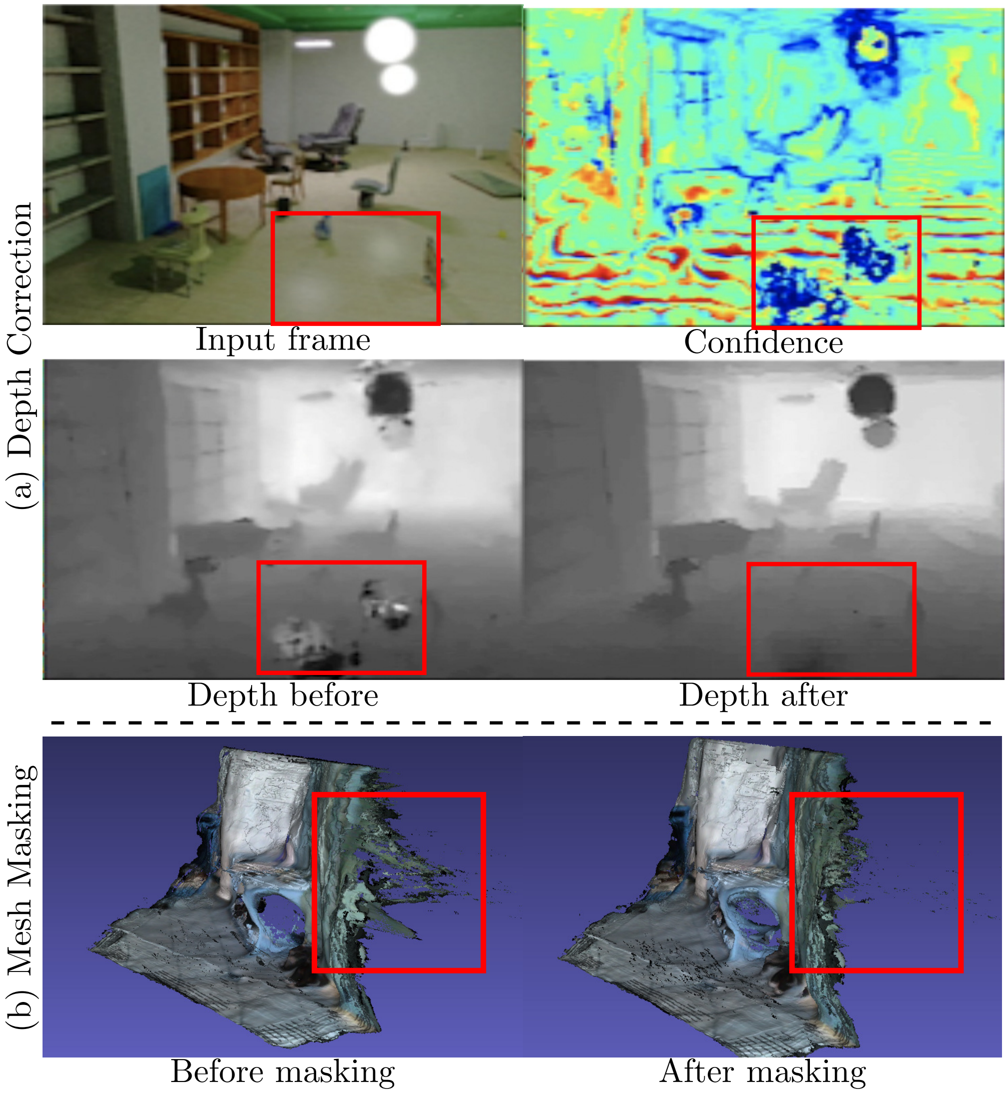


Figure 4.13: Usefulness of depth confidence map. (a) Correct depth map using Fast Bilateral Solver [8]. (b) Mask out pixels with low confidence before applying Voxel Hashing [137].

4.5 Limitations

There are several limitations that we plan to address in the future. First, camera poses from a monocular video often suffer from scale drifting, which may affect the accuracy of our depth estimation. Second, in this work we focus on depth sensing from a local time window, rather than solving it in a global context using all the frames. In the future, we plan to integrate our method into visual SLAM systems to correct drifting and further improve depth quality.

4.6 Conclusions

In this chapter, we present a DL-based method for continuous depth sensing from a monocular video camera. Our method estimates a depth probability distribution volume from a local time window, and integrates it over time under a Bayesian filtering framework. Experimental results show our approach achieves high accuracy, temporal consistency, and robustness for depth sensing, especially for the cross-dataset tasks. The estimated depth maps from our method can be fed directly into RGB-D scanning systems for 3D reconstruction and achieve on-par or sometimes more complete 3D meshes than using a real RGB-D sensor.

Chapter 5

See below human skin with EpiVerge

5.1 Introduction

Imaging below the skin and through tissue is important for diagnosis of several dermatological and cardiovascular conditions. MRI remains the best current approach to obtain a 3D dimensional visualization below the skin. But MRI is expensive, requires visits to a hospital or imaging center, and the patients are highly inconvenienced. Non-invasive imaging using visible or near-infra-red light has the potential to make devices portable, safe, and convenient to use at home or at point-of-care centers.

While light penetrates deep through tissue, it scatters continuously resulting in poor image contrast. This makes it challenging to recover useful properties about the anatomical structures below the skin. Further, the anatomical structures include a complex heterogeneous distribution of tissue, vasculature, tumors (benign or malignant) that vary in optical properties (density, scattering, absorption) and depths below the skin. This makes the modeling of light propagation below skin challenging.

Fortunately, under the highly scattering regime, the photons can be assumed to be traveling diffusely in the medium and can be described as a random walk. This has enabled accurate forward models under diffuse photon propagation. In order to improve contrast, imaging detectors and sources are placed at different locations on the skin. This arrangement captures only indirectly scattered light while eliminating the dominant direct reflection and backscatter [1]. The approaches that attempt to invert the diffusion model with such indirect light imaging systems are commonly classified as ”Diffuse Optical Tomography” (DOT). Due to their portability and ease of use, DOT is becoming an attractive choice over traditional modalities like MRI for cerebral as well as hemodynamic imaging [139, 197]. More recently, DOT has been shown to be a promising tool in detecting strokes [22], breast cancer [68], and thyroid imaging [61].

But there are two important drawbacks to existing DOT approaches. First, the number of source-detector pairs limits the form-factor of devices built so far. Even with multiple source-detector pairs, applying traditional inverse methods for DOT results in poor resolution, as shown

[1] Analogously, in vision and graphics, works measure the Bi-directional Sub-surface Scattering Reflectance Distribution Function (BSSRDF) [45, 46, 59, 90, 188]

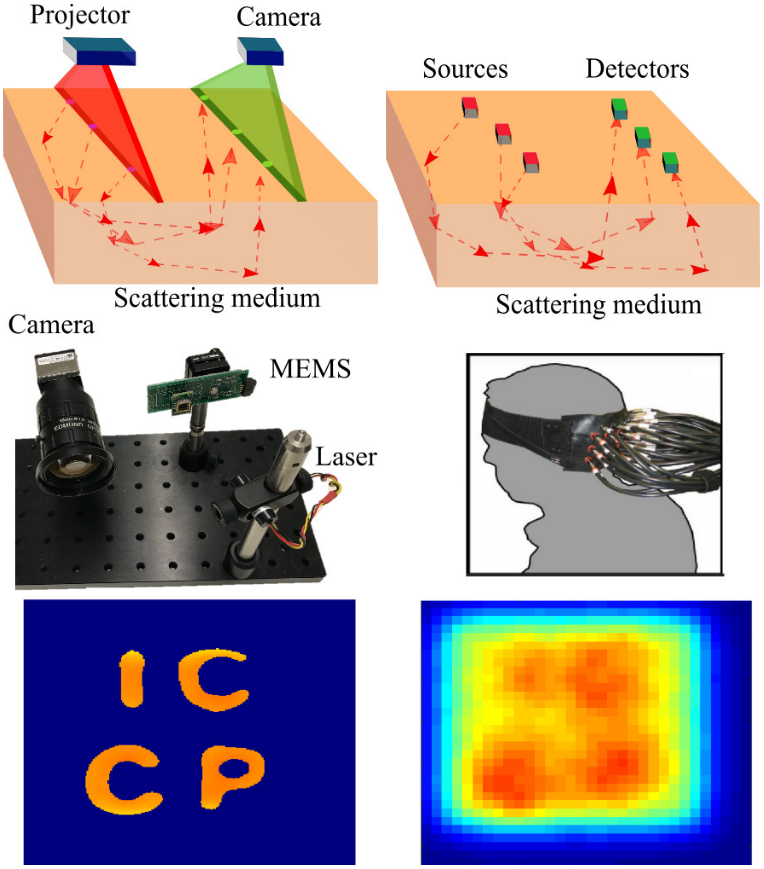


Figure 5.1: Diffuse Optical Tomography (DOT) with line-scanned camera and line scanning MEMS projector (left) compared with traditional DOT [2] with 25 source-detector pairs (right). Both arrangements capture short range indirect subsurface scattered light but our approach is more efficient and recovers the medium (bottom row) at much higher resolution.

in the second column of Figure 5.1. Second, as the number of source-detector pairs increases, the computational complexity of the algorithms that recover the volumetric medium increases prohibitively [20]. In this chapter, we present an imaging and algorithmic approach to resolve these fundamental drawbacks in DOT systems. Instead of separate detector-source pairs, we use a high resolution 2D camera and a MEMS projector to obtain a large number of effective source-detector pairs, as is commonly done in vision and graphics. This makes the DOT system much more compact and programmable. Second, to increase the speed of acquisition, we illuminate a line on the skin and capture a different line in the sensor, as described in [101]. This arrangement captures short-range indirect light transport much faster than point-to-point illumination and sensing. But [101] uses a rectified configuration where the projector and camera planes are parallel [141], leading to a low spatial resolution over a large overlapping stereo volume. We develop a new design with a verged configuration that enables high spatial resolution imaging within a small region on the skin (approximately 8 cm x 8 cm).

Using this verged design, we develop an efficient algorithm that is based on the convolution approximation to the forward model for light propagation in a heterogeneous subsurface medium. We show that the convolution kernel is independent of the heterogeneous structure and only depends on the imaging configuration and the scattering properties of the homogeneous scattering medium. This allows us to recover the heterogeneous structures at much higher spatial resolution compared with the traditional DOT, as shown in the last row of Figure 5.1.

We evaluate our imaging and algorithmic approaches on simulated data by borrowing optical parameters of skin, tissue, blood and vasculature from bio-optical literature [108]. We then demonstrate our approach with an imaging setup that consists of a high resolution 2D camera and a custom-built MEMS based laser projector that are verged to capture near-microscopic spatial resolution beneath a small area of the surface. This imaging system is used to recover heterogeneous structures immersed within a highly scattering medium, such as milk. We show that with the proposed hardware and algorithm, we can detect reasonably accurate boundaries of structures up to a depth of $8mm$ below the surface of milk. We believe this work represents strong progress in achieving high-spatial resolution diffuse optical tomography for the first time at subsurface depths of several millimeters in highly scattering media.

5.2 Related Works

Over the past few years, there have been developments mainly on two aspects of DOT - improving the instrument system design [167, 169, 219], and secondly, on theoretical aspects that involves accurate forward modeling and rendering [6, 13]. The traditional DOT setup consists of illumination sources and detectors which are placed on the tissue or skin surface. To improve the reconstruction of optical parameters of the tissue volume, multiple configurations of DOT have been explored. In general, it is not possible to obtain depth information from a single source-detector system and therefore multiple source-detector configurations are needed [165].

More recent systems utilize the time-domain (TD) information of photon propagation. TD-DOT systems consist of a source emitting a pulse of light and fast-gated detectors that capture the time-profile of photon arrival. The detectors and the sources are located on a probe with fixed strategic distances between them, so that the photons have traversed a certain depth and rejecting early arriving photons [149]. The most important drawback in these systems is that the limited number of detectors constrains the spatial resolution of the reconstructed optical parameters. More recent DOT systems use structured illumination, which involves projection of patterns instead of discrete sources [10, 66, 100, 152]. The light after interaction with the tissue is captured by either a single-pixel detector or a 2D CCD camera. The use of structured illumination addresses the issues of low speed and sparse spatial sampling, which are associated with traditional DOT systems [152].

The reconstruction of optical parameters involves fitting a forward model to the acquired measurements. The forward model can be obtained either from mesh-based Monte-Carlo simulations [50, 215] or from a diffusion approximation [16, 17, 48] derived from radiative-transfer equations (RTE). While the Monte-Carlo based forward model is more accurate, it requires hours of computing time. Under Born or Rytov approximations, the forward model relates the optical parameters and the measurements by a linear set of equations [140]. Generally, the number of optical parameters to be reconstructed per voxel is very large compared to the number of measurements, and therefore the inverse problem is severely ill-posed. Tikhonov regularizer or sparsity-inducing regularizers are commonly applied for solving the inverse problem [30, 75]. However, with dense sampling the computational load increases as the reconstruction process involves inversion of a large-scale Jacobian matrix [213].

A notable alternative approach [218] is to use the Monte Carlo estimator to differentiate the RTE with respect to any arbitrary differentiable changes of a scene, such as volumetric scattering property, anisotropic phase function or location of heterogeneity. This approach shares the same generality as RTE. However, the performance of the differentiation-based method is highly dependent on the initial estimation of the variables and tend to be trapped in local minimal. Our conventional formulation simplifies the RTE such that the inverse problem is convex. As a result, the result of our method can be used as a good initial guess for the full RTE method [218].

5.3 Forward model

In this section, we will review the derivation of the basic theory in DOT for dense scattering tissues. First, we will derive the Diffusion Equation for the surrounding homogeneous medium from the Radiative Transfer Equation [21, 81], assuming that the homogeneous scattering medium surrounding the heterogeneities is dense enough such that the light propagation in the medium is dominated by the high-order light scattering events and the angular distribution of light propagation direction is isotropic. Then we will derive the Diffusion Equation for the heterogeneities, assuming that the light absorption coefficient discrepancy dominates the scattering property difference between the heterogeneous embedding and the surrounding medium. Although the assumptions do not always hold perfectly, we find that our proposed method is robust to the cases where the assumptions fail through evaluations in Section 5.6.1.

The Radiative Transfer Equation (RTE) describes the light radiance, which models light propagation, at a particular position in the scattering medium at a specific time instant. It is generally difficult to solve the RTE in closed form. When the medium is highly scattering, as in the case of biological tissue, the diffusion approximation is commonly applied to obtain the diffusion equa-

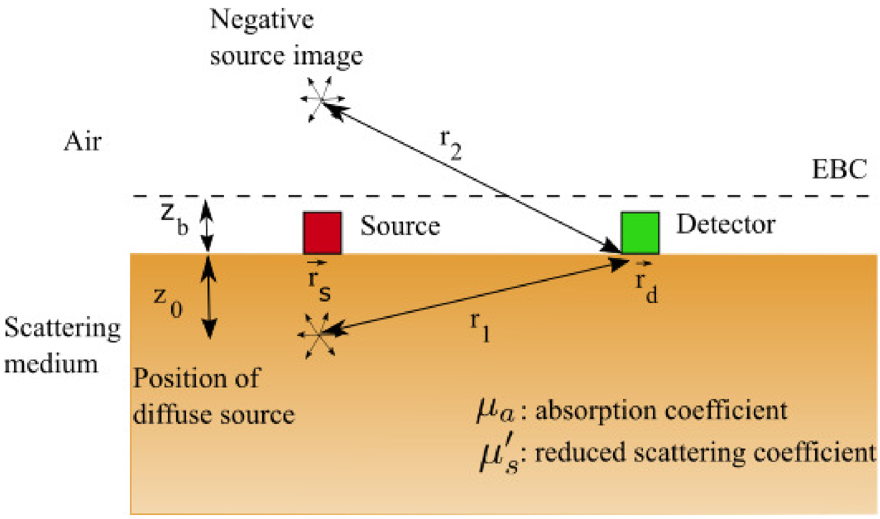


Figure 5.2: Source-detector configuration in typical DOT system. The fluence rate at the detector is given by superposition of the real diffuse source located z_0 below the surface, and a negative image source around the zero flux line denoted by EBC.

tion [48, 90]. The photon diffusion equation models the fluence rate Φ, that is defined as the total light radiance integrated over all directions, at a position $\vec{r}$ in the scattering medium for a continuous intensity light source S, given as,

$$(-D(\vec{r})\nabla^2 + \mu_a(\vec{r}))\Phi(\vec{r}) = S(\vec{r}), \tag{5.1}$$

where $\mu_a(\vec{r}), \mu'_s(\vec{r})$ are the absorption coefficient and the reduced scattering coefficient of the medium respectively, and $D(\vec{r}) = 1/(3(\mu'_s(\vec{r}) + \mu_a(\vec{r})))$ is known as the diffusion coefficient of the scattering medium. The tissue is commonly modeled as a semi-infinite scattering homogeneous medium, with the source and the detector positioned at the air-medium interface. When the light source is treated as a constant pencil beam source, i.e. $S(\vec{r}) = S\delta(\vec{r}_s)$, the solution for fluence rate in (5.1) for the configuration in Figure 5.2 can be written in a analytical form using extrapolated zero boundary conditions (EBC):

$$\Phi_0(\vec{r}_d, \vec{r}_s) = \frac{S}{4\pi D_0}\left[\frac{e^{-\beta r_1}}{r_1} - \frac{e^{-\beta r_2}}{r_2}\right], \tag{5.2}$$

where, $\Phi_0(\vec{r}_d, \vec{r}_s)$ is the fluence rate at detector kept at a position $\vec{r}_d$ with a constant source at $\vec{r}_s$ [17]. The diffusion coefficient of the homogeneous medium is denoted by D_0 and the term $\beta = \sqrt{3\mu'_s\mu_a}$ depend on the optical properties of the homogeneous scattering medium. The extrapolated boundary condition (EBC) accounts for the refractive index mismatch of air and the medium. Solving for the boundary condition defines a zero fluence rate line located z_b above the air-medium interface. This boundary line is imposed by placing a negative image of the source around the zero-crossing line [17]. The terms r_1 and r_2 are the distances from the detector to the real and the negative image source respectively, and they are defined as:

$$r_1 = |\vec{r}_s + z_0 - \vec{r}_d|,$$
$$r_2 = |\vec{r}_s - z_0 - 2z_b - \vec{r}_d|, \tag{5.3}$$

where, $z_0 = 3D$ is the location of diffused source in the medium. The term z_b is the distance of the zero fluence rate boundary from the air-medium interface.

Often, we are interested in reconstructing objects like veins or tumors embedded within human tissue. Typically these objects have different optical parameters than the background medium. In the presence of heterogeneity, the change in absorption coefficient of the medium can be defined as,

$$\mu_a(\vec{r}) = \mu_{a_0} + \delta\mu_a(\vec{r}) \tag{5.4}$$

where $\delta\mu_a(\vec{r})$ is the difference in absorption coefficient of the heterogeneous object over the background medium. We assume that the change in the reduced scattering coefficient $\mu_s'(\vec{r})$ is negligible and can be ignored. The resultant fluence rate at the detector position $\vec{r}_d$ for a source at $\vec{r}_s$ is written as a linear addition of fluence rate from homogeneous component $\Phi_0(\vec{r}_d, \vec{r}_s)$ and the change in fluence rate $\Delta\Phi(\vec{r}_d, \vec{r}_s)$ due to the heterogeneous object,

$$\Phi(\vec{r}_d, \vec{r}_s) = \Phi_0(\vec{r}_d, \vec{r}_s) + \Delta\Phi(\vec{r}_d, \vec{r}_s). \tag{5.5}$$

The change in fluence rate is due to the absorption coefficient change across the volume around the point of interest [17]:

$$\Delta\Phi(\vec{r}_d, \vec{r}_s) = -\int \Phi_0(\vec{r}_s, \vec{r}_j)\frac{\delta\mu_a(\vec{r}_j)}{D_0}G_0(\vec{r}_j, \vec{r}_d)d\vec{r}_j, \tag{5.6}$$

where G_0 represents the Green's function for a homogeneous slab and is related to Φ_0 in (5.2) as $G_0 = D_0\Phi_0/S$.

We acquire images using a CCD camera, which records the radiant exitance at the surface. The radiant exitance is proportional to the diffuse reflectance R, which is the projection of current density along the surface normal of the detector for a unit-power source,

$$R(\vec{r}_d, \vec{r}_s) = D_0\left[\frac{\delta\Phi}{\delta z_d}\right]_{z_d=0}, \tag{5.7}$$

where z_d is the z component of the detector location $\vec{r}_d$.

Applying a derivative to (5.5) with respect to z_d and multiplying by D_0, we obtain,

$$R(\vec{r}_d, \vec{r}_s) = R_0(\vec{r}_d, \vec{r}_s) + \Delta R(\vec{r}_d, \vec{r}_s), \tag{5.8}$$

where $R_0 = D_0\left[\delta\Phi_0/\delta z_d\right]_{z_d=0}$ is the diffuse reflectance due to the homogeneous background medium and is obtained by taking a derivative of Φ_0 in (5.2) with respect to z_d, given by [90],

$$R_0 = \frac{1}{4\pi}\left[\frac{z_0(1+\beta r_1)e^{-\beta r_1}}{r_1^3} + \frac{(z_0+2z_b)(1+\beta r_2)e^{-\beta r_2}}{r_2^3}\right] \tag{5.9}$$

Similarly, ΔR represents the change in diffuse reflectance for the heterogeneous object. The expression for ΔR is obtained by taking a derivative of (5.6) with respect to z_d and multiplying by D_0, resulting in the following expression,

$$\begin{aligned}\Delta R(\vec{r}_d, \vec{r}_s) &= -\int \Phi_0(\vec{r}_s, \vec{r}_j)\delta\mu_a(\vec{r}_j)\left[\frac{\delta G_0(\vec{r}_j, \vec{r}_d)}{\delta z_d}\right]_{z_d=0} d\vec{r}_j, \\ &= -\int \Phi_0(\vec{r}_s, \vec{r}_j)\delta\mu_a(\vec{r}_j)R_0(\vec{r}_j, \vec{r}_d)d\vec{r}_j.\end{aligned} \tag{5.10}$$

We discretize the integral above by dividing the medium into N voxels, and the optical properties are defined for each voxel. If the medium is discretized into N voxels with volume of each voxel as h^3, then (5.10) can be written in the discrete summation form given by

$$\Delta R(\vec{r}_d, \vec{r}_s) = -\sum_{j=1}^{N} P(\vec{r}_s, \vec{r}_j, \vec{r}_d)\delta\mu_a(\vec{r}_j), \tag{5.11}$$

with

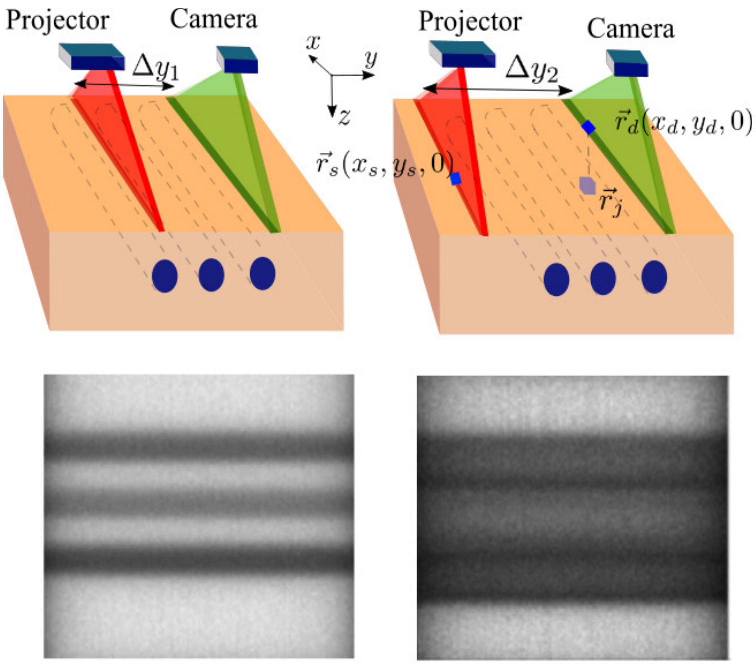
$$P(\vec{r}_d, \vec{r}_j, \vec{r}_s) = \Phi_0(\vec{r}_s, \vec{r}_j)R_0(\vec{r}_j, \vec{r}_d)h^3. \tag{5.12}$$

The term $P(\vec{r}_s, \vec{r}_j, \vec{r}_d)$ is commonly known as the phase function defined at each voxel position $\vec{r}_j$ in the medium. The values of the phase function depend on the optical properties of the background homogeneous medium as well as the location of the source $\vec{r}_s$ and the detector $\vec{r}_d$. Note that the phase function is independent from the structure of the heterogeneous object.

5.4 Convolution approximation of heterogeneous model

In this section, we describe how the diffuse forward model can be adapted to our experimental setup. We project a line illumination on the scene using a laser projector as in [101]. So the light source is now considered as a slit source instead of a point source. By using a slit source we reduce the acquisition time since line scanning is significantly faster than point scanning. We incorporate a slit source in the forward model by using the linear superposition principle. The quantities described in the previous section which are functions of the source location $\vec{r}_s$ are now obtained by adding up the contributions corresponding to all the point sources on the illumination line.

On the detector side, we use a rolling shutter camera synchronised with the laser projector. The advantage of using a camera is that each pixel in the camera sensor can now be considered as an individual detector, and hence we have a detector array with millions of detectors. Secondly, since the camera sensor can be considered as a grid array of detectors, we can derive a convolution form of the forward model, significantly speeding up the computation time. We acquire several images by varying the pixel to illumination line distance shown in Figure 5.3. These images are referred to as short-range indirect images. The boundaries of the heterogeneous structures become more blurry in the short range indirect image as the pixel to illumination line distance Δy increases. The blurring effect is related to Δy and the depth of the structures. This is similar to the depth from



(a) (b)

Figure 5.3: Generation of short range indirect images for a small (a) and a large (b) pixel to illumination distance Δy. The simulated scene consists of three cylinders embedded in a scattering medium. The offset Δy is kept constant while scanning the entire scene to obtain an image. For a shorter Δy as in (a), the rods are distinctly visible, whereas for longer Δy, the blurring increases with reduction of signal-to-noise ratio.

(de)focus methods, where the blurring effect is related to the focal setting of the camera and the scene depth.

The values of phase function at each voxel for the short-range indirect images can be interpreted as the number of photons that have traversed through the corresponding voxel for a given pixel to illumination line distance. Typically, the most probable path between a pixel and the source illumination line follows a well-known ”banana shape” [37] and is shown for different pixel to illumination line distances in the Figure 5.4.

We note two important properties of the phase function P. Firstly, in case of simple geometry like the semi-infinite homogeneous background medium under consideration, the expression for the Green’s function G_0 as well as Φ_0 can be written in terms of relative voxel location rather than the absolute location, i.e,

$$\begin{aligned} P(\vec{r}_d, \vec{r}_j, \vec{r}_s) &= \Phi_0(\vec{r}_s - \vec{r}_j) R_0(\vec{r}_j - \vec{r}_d), \\ &= P(\vec{r}_j - \vec{r}_d, \vec{r}_s - \vec{r}_d). \end{aligned} \tag{5.13}$$

Secondly, we note that the values of the phase function decays fast spatially as the distance between a voxel and source or detector position increases. Hence, we can neglect the contribution of voxels that are far away from both the illumination line and the pixel. Since we are using

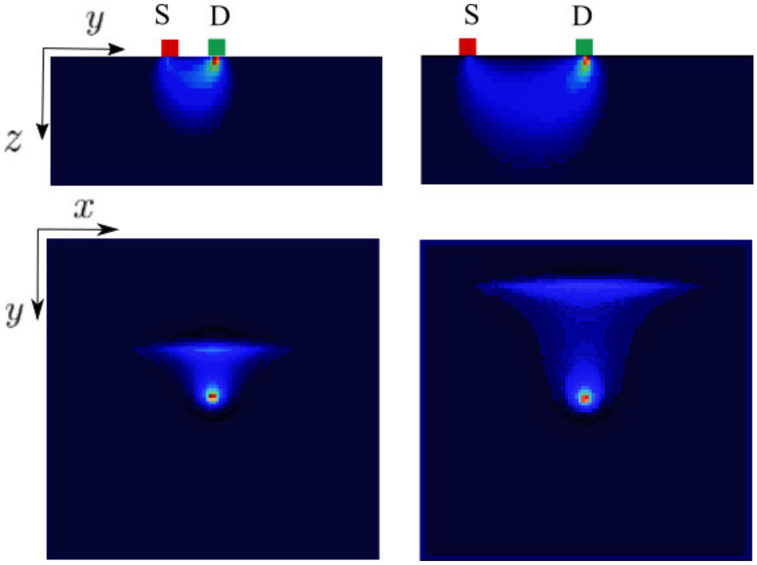


Figure 5.4: Visualization of phase function for different pixel to illumination line distance in y-z plane (top row), and x-y plane (bottom row). S and D represents the illumination line and pixel location respectively. As the pixel to illumination line distance increases, the photons tend to travel deeper into the scattering medium but leads to reduced number of photons reaching the pixel, thereby reducing the signal-to-noise ratio.

a projected line illumination as our light source, we approximate the phase function in (5.13) by the summation of truncated phase function for each source point along the illumination line. Additionally, as evident from the figure, the contribution of light from the illumination line to a center pixel is dominant only near the center of the illumination line, and hence we can use a spatially-invariant phase kernel κ. We define the pixel to illumination line distance $\Delta y = y_s - y_d$, where y_s and y_d are the y component of illumination row $\vec{r}_s$ and the pixel location $\vec{r}_d$ respectively. The phase kernel for a line illumination can then be written as,

$$\kappa(\vec{r}_j - \vec{r}_d; \Delta y) = \sum_{\vec{r}_s} P(\vec{r}_j - \vec{r}_d, \vec{r}_s - \vec{r}_d), \tag{5.14}$$

where the summation over $\vec{r}_s$ is for all the point sources lying on the illumination line. In the following, we will denote the phase kernel as $\kappa(\Delta y)$ for denotation simplicity unless the spatial dependency needs to be emphasized.

Similarly, the diffuse reflectance $R(\vec{r}_d, \vec{r}_s)$, the change in diffuse reflectance $\Delta R(\vec{r}_d, \vec{r}_s)$ and the homogeneous diffuse reflectance $R_0(\vec{r}_d, \vec{r}_s)$ in (5.8) are modified for line illumination as the sum of contribution from all point sources lying on the illumination line, and are defined as $R(\vec{r}_d; \Delta y)$, $\Delta R(\vec{r}_d; \Delta y)$ and $R_0(\vec{r}_d; \Delta y)$ respectively. We denote (x_d, y_d) as the surface coordinates of the pixel location $\vec{r}_d$ as shown in Figure 5.3. If the change in absorption coefficient $\delta\mu_a(\vec{r}_j)$ in (5.11) is represented by a 3D volume Q, then the change in diffuse reflectance ΔR in (5.11) can now be expressed in a convolution notation as

$$
\begin{aligned}
\Delta R(x_d, y_d; \Delta y) &= -\sum_{\vec{r}_s} \sum_{j=1}^{N} P(\vec{r}_j - \vec{r}_d, \vec{r}_s - \vec{r}_d) \delta\mu_a(\vec{r}_j) \\
&= -\sum_{j=1}^{N} \kappa(\vec{r}_j - \vec{r}_d; \Delta y) \delta\mu_a(\vec{r}_j)
\end{aligned} \tag{5.15}
$$

where $\Delta R \in \mathbb{R}^{M \times N}$ is defined over a sensor array of dimension $M \times N$ and corresponds to each pixel to illumination line distance Δy as shown in Figure 5.3. By representing the change of absorption coefficient $\delta\mu_a$ by a 3D volume Q, we can rewrite (5.15) as the sum of a 3D convolution results:

$$
\Delta R(x_d, y_d; \Delta y) = -\sum_{z} \kappa(\Delta y) * Q(x_d, y_d, z) \tag{5.16}
$$

The change in absorption coefficient in the 3D volume is denoted by $Q \in \mathbb{R}^{M \times N \times D}$, where D is the depth resolution. The 3D truncated kernel $\kappa \in \mathbb{R}^{m \times n \times D}$ is the defined for each Δy, and has the same depth resolution as that of the 3D volume Q. Using (5.8), the resultant diffuse reflectance R acquired at each pixel to illumination line distance Δy can be written as a linear summation of the contribution from homogeneous background medium R_0 and the perturbation term due to presence of heterogeneity ΔR,

$$
R(x_d, y_d; \Delta y) = R_0(x_d, y_d; \Delta y) - \sum_{z} \kappa(\Delta y) * Q(x_d, y_d, z)
$$

where $R \in \mathbb{R}^{M \times N}$ is the diffuse reflectance on an $M \times N$ grid.

5.4.1 Reconstruction of heterogeneous structure

For the set of captured images which correspond to different pixel to illumination line Δy, we capture a set of short range indirect images $I(\Delta y)$. For the given set of images, we reconstruct the volume Q of unknown optical parameters by solving the following optimization problem,

$$
\min_{Q} . \sum_{\Delta y = T_{d_{min}}}^{T_{d_{max}}} ||I(\Delta y) - l(R_0(\Delta y) - \kappa(\Delta y) * Q)||_F^2 + \lambda ||Q||_1, \tag{5.17}
$$

where $||.||_F$ denotes the Frobenius norm, and l is an unknown scaling factor which depends on the intensity and width of the laser profile and the sensitivity of the camera. The procedure for determining this factor l is highlighted in more detail in Section 5.6.2. We also assume the reconstructed volume to be sparse, which essentially implies that the heterogeneous object only occupies a fraction of the total reconstructed volume.

The optimization is done over a range of Δy values. For smaller Δy values, the diffusion approximation breaks down, as the photon propagation is largely governed by single or very few scattering events. For very large Δy, not enough photons reach the camera pixels, and therefore the measurement images have a poor signal-to-noise ratio. Therefore, the range of Δy values needs to be chosen appropriately.

If we know the exact optical parameters μ_s' and μ_a of the homogeneous background medium, then we can construct the kernel $\kappa(\Delta y)$ as in (5.14). However in some cases, the background optical parameters of the material are not known. In those cases, we select a homogeneous patch inside the field of view, and fit the pixel intensity measurements with lR_0 with respect to the unknown optical coefficients as in (5.9). We then use the estimated values of the coefficients to construct the phase kernel $\kappa(\Delta y)$ for solving the optimization in (5.17).

We use PyTorch for implementation given it is highly optimized for convolution operations. The running time on a workstation with TianX GPU is around 5 minutes for 300 iterations for Q with a depth resolution of 64. The λ value in (5.17) is heuristically chosen to be 0.0001. We start the optimization with an initial value of all zeros for Q, and the reconstruction accuracy can be further improved if a better initialization is provided based on prior knowledge of the scene.

5.5 Hardware

In this section, we describe our imaging setup for capturing short-range indirect images. In [101], a rectified configuration where the projector and camera are parallel is used for capturing the short-range images. That setup leads to a low spatial resolution over a large overlapping stereo volume. To capture high resolution images for small area of interest, we need a high spatial resolution over a smaller overlapping stereo volume. One way to achieve smaller overlapping stereo volume is to verge the projector and camera. This motivates us to design a verged setup for capturing high resolution short-range indirect images.

Our setup consists of a pair of synchronized rolling shutter camera and a laser projector implemented with Micro-Electro-Mechanical-Systems (MEMS). Our imaging setup is shown in Figuer 5.1 and Figure 5.7 (a). We use IDS-3950CPv2 industrial camera and Mirrorcle MEMS development kit. The central wavelength for the laser light is 680 nm. The MEMS mirror reflects the laser beam from the laser diode and the orientation of the MEMS mirror can be controlled in terms of two rotation axes (vertical and horizontal). The size of the imaged area on the sample is 8 cm by 8 cm. We model the laser diode and MEMS mirror pair as a pinhole projector whose center of projection is the center of rotation of the MEMS.

During the imaging process, the projector is scanned through the epipolar planes of the projector-camera pair. The camera is synchronized such that the pixels having a pre-defined offset from the corresponding epipolar line on the camera image plane are exposed. Each offset corresponds to one pixel to illumination line distance Δy as discussed in Section 5.4. For the physically rectified projector-camera pair as in [101], the epipolar lines on the projector and camera image are horizontal. This simply corresponds to illuminating and exposing the corresponding rows of projector and camera. In contrast, in our setup, the epipolar lines in the projector and camera are not horizontal due to the verged setup. So we cannot capture the short range indirect images by illuminating and exposing corresponding rows. Instead, on the projector side, we control the angle of the MEMS mirror to scan the light laser beam across a pencil of epipolar lines with different 2D slopes in the projector image plane. On the camera side, we interpolate over offset epipolar lines to get the

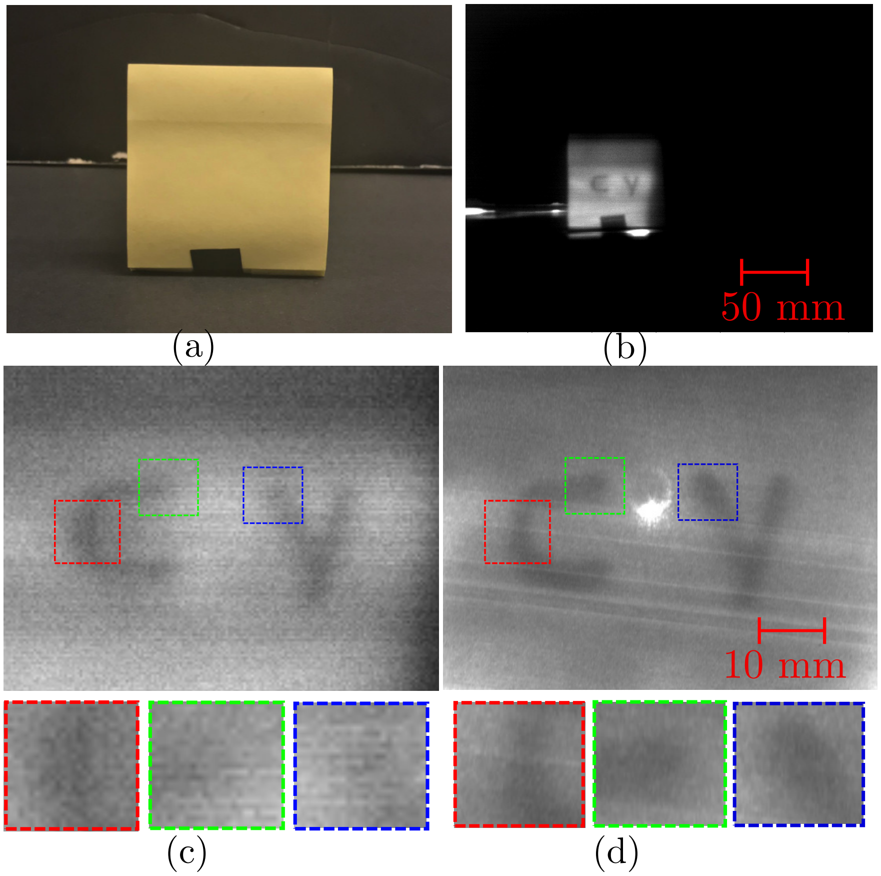


Figure 5.5: Images of a paper sticker captured using different devices. The sticker page with letters is occluded by several other pages so no letters can be seen under regular lighting. (a) Image captured with cellphone camera under regular lighting. ; (b) Short-range indirect image captured with the device in [101]; (c) Enlarged image for the sticker region in (b); (d) Short-range indirect image captured with our device. Our device has smaller FOV due to non-zero vergence angle. The images captured with our device has higher resolution, SNR and contrast as shown in the insects in (c) and (d). The bright spot in the center and the lines in (d) is due to the reflection and inter-reflections from the protective glass in front of the MEMS mirror.

short range indirect images. As a special case, for $\Delta y = 0$, the interpolated line overlaps with the epipolar. The resultant image is the direct light image.

Our image setup has smaller FOV than the rectified system in [101] due to the non-zero vergence angle between the project and camera. As a result, we can place the sample closer to the camera while the sample can still be illuminated by the projector. This enables higher image resolution for smaller area of interest so that more fine-grained (sub)surface details can be captured. In Figure 5.5, we show the images of a paper sticker captured with different devices. The sticker page with letters is occluded by several other pages so no letters can be seen under regular lighting. The occluded letters are visible in the short range indirect images from both [101] and our setup. Our device has smaller FOV and higher spatial resolution over the region of interest due to the verged configuration. In addition, we have better contrast and higher SNR because the laser light source used in our setup is of higher intensity compared to the pico-projector in [101]. The bright spot in the center and the lines in (d) is due to the reflection and inter-reflections from the protective glass in front of the MEMS. Due the higher image resolution enabled by the verged setup, more structure details of the subsurface structure, such as blood veins underneath human skin can be observed with our proposed setup, as show in Fig. 5.6.

5.5.1 Calibration

The device is mounted vertically above the liquid container as shown in Figure 5.7 (a), with no cover above the scattering medium. We model the laser-MEMS as a pinhole projector whose center of projection is the rotation center of the MEMS mirror. During the calibration process, we estimate the relative pose between the MEMS mirror and the camera. For MEMS, we compensate for the non-linear mapping between the input voltage for the MEMS and the mirror tilt angle, and account for the mis-alignment of the MEMS mirror and the laser, as shown in Figure 5.7 (b).

More specifically, we illuminate planes with given poses relative to the camera with a set of dot patterns. As shown in Figure 5.8, given the laser dot images for different plane positions, we can fit the laser rays in 3D and triangulate the rays to get the origin of the rays, *i.e.* the rotation center of the MEMS mirror. Due to the laser-MEMS misalignment and fitting error for the rays, the rays will not intersect at one 3D point. We solve a least square problem for the intersection point where the point to ray distances are minimized. The fitted rays are also used to account for the non-linear relation between the MEMS input voltage and the rotation angle. In calibration, we build a lookup table relating the input voltage for the MEMS and the rotation angle for the mirror to account for their non-linear relation. During imaging, given the target laser ray direction, we can estimate the required input voltage by interpolating over lookup table.

5.6 Experiment Results

5.6.1 Simulation

We test the proposed algorithm using Monte Carlo rendered images. For the homogeneous medium, we use the scattering coefficients of human skin measured in [90]. The heterogeneous inclusions

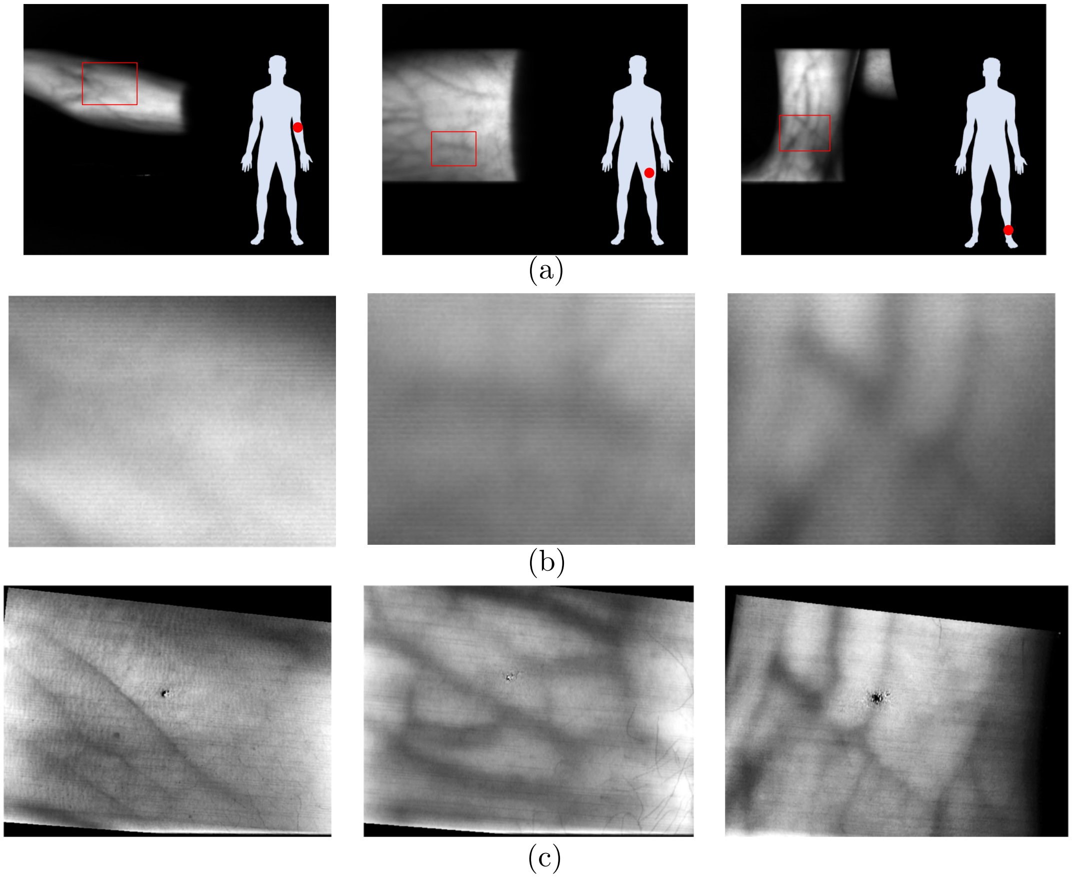
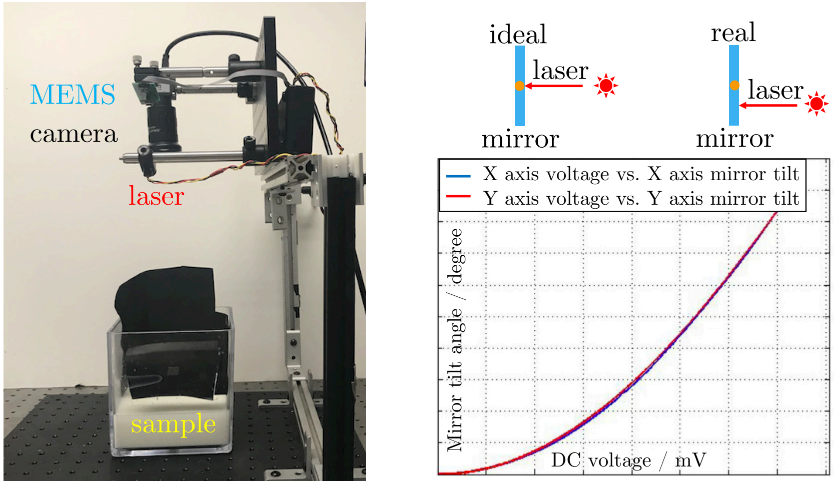
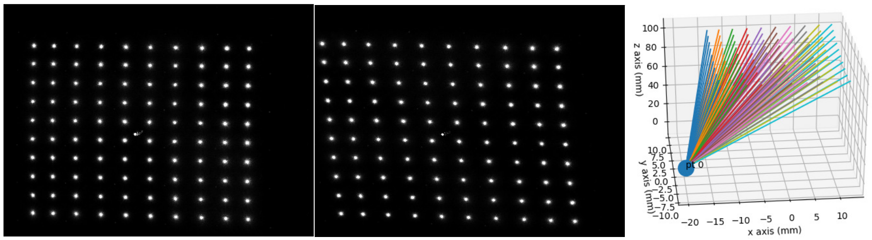


Figure 5.6: Subsurface imaging for human skin. (a)(b) The image and zoomed-in region of the human skin, captured with the imaging system from [101]. (c) The image of the same body region captured with our device. More vein details are visible, at the expense of smaller FOV.


MEMS
camera
laser
sample
ideal
laser
mirror
real
laser
mirror
X axis voltage vs. X axis mirror tilt
Y axis voltage vs. Y axis mirror tilt
Mirror tilt angle / degree
DC voltage / mV

(a) Experiment setup (b) MEMS calibration

Figure 5.7: Experiment setup and calibration to compensate the laser-mirror misalignment and non-linearity of MEMS. (a) The device is mounted vertically above the sample container, with no cover above the scattering medium. (b) During MEMS calibration, we consider the misalignment between the laser and mirror (above) and non-linearity of MEMS mechanics. Due to misalignment, the incident laser beam onto the MEMS mirror will not be perpendicular to the mirror surface and align with the MEMS rotation center; Due to non-linearity of MEMS mechanics, the input control signal and degrees of rotation are not linearly related.



(a) (b) (c)

Figure 5.8: The MEMS-camera pose calibration. We illuminate a plane with known pose with 2D array of beams as shown in (a) and (b). Given the plane orientations, we can get the 3D parameters for the rays from multiple images. Then we triangulate all the fitted rays to determine the center of projection for the projector, in our case, the rotation center for the MEMS mirror.

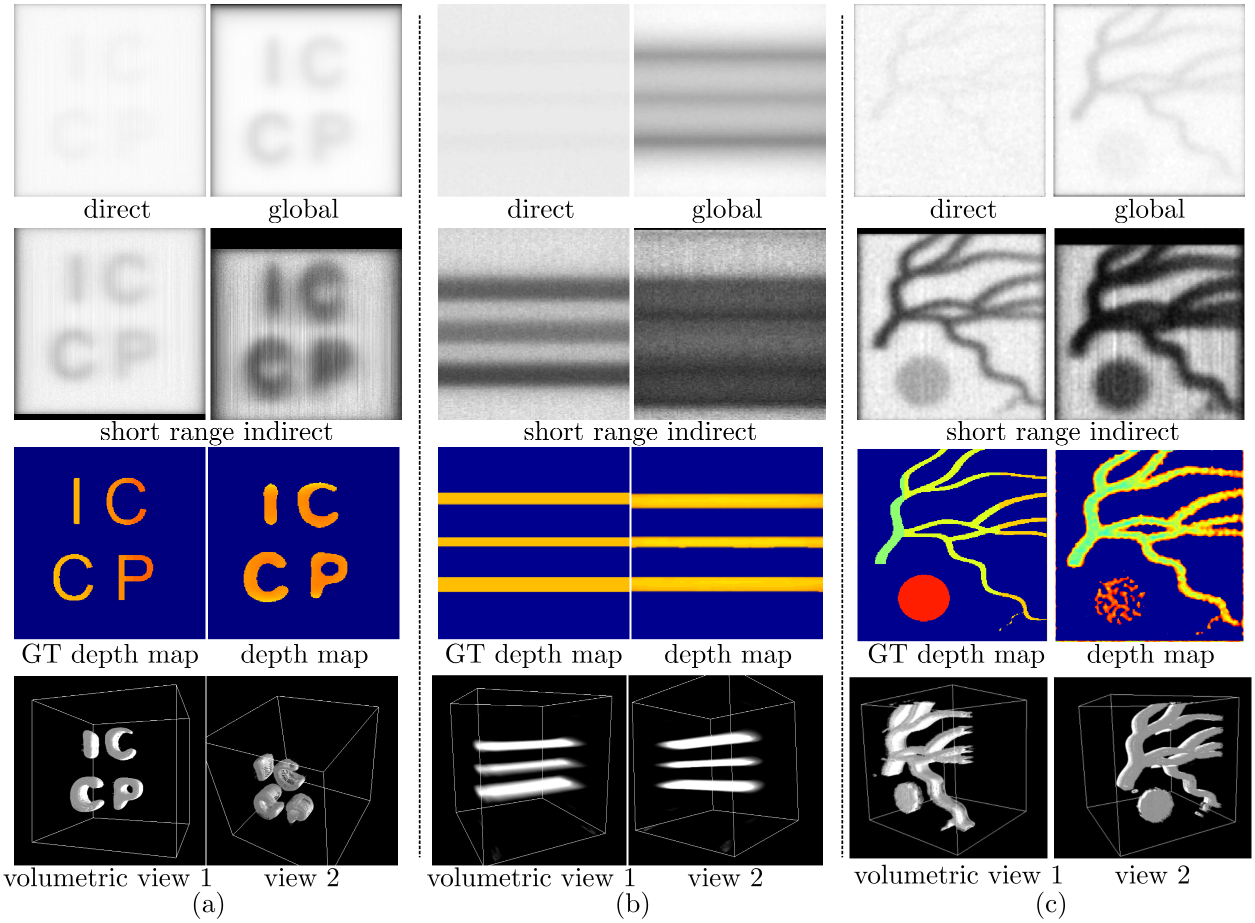


Figure 5.9: Simulated direct/global light image, the short range images with different Δy settings, and the DOT results. The homogeneous medium is skin in (a) and (c), skim milk in (b), with the scattering coefficients measured in [90]. Photon and read-out noise are added to the input images. The depths of inclusions in (a) and (b) are $4mm$ and $3mm$ respectively. In (c), the depth of the vein structure is $5mm$ while the solid circle is $10mm$ deep, The inclusion boundaries in the global and short-range indirect images are either blurred or shifted due to light scattering. The signal-to-noise ration decreases as the pixel to illumination line distance increases since less photons are received by the detectors. Our methods recovered the inclusion boundaries and their relative depths despite the blurring and noises.

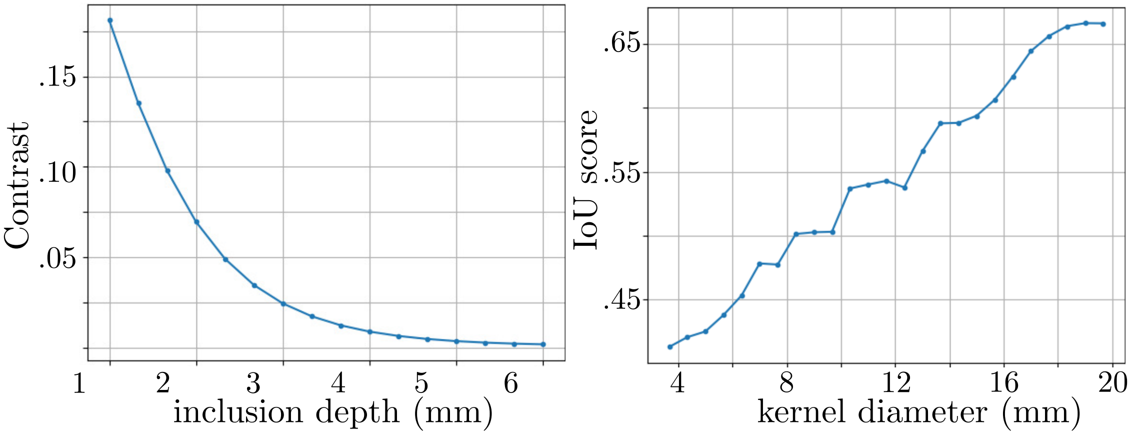


Figure 5.10: (a) The average contrast of the short range indirect image varies with the inclusion depth. The inclusion depth is the distance between the inclusion and the embedding medium surface. (b) IoU vs. kernel sizes. For human skin, the performance saturates when the kernel size approaches 20 mm in diameter For all simulations, the images are synthesized using Monte Carlo method with scattering properties for human skin measured in [90].

are located up to 4 mm below the surface. For the imaging setup, the width of the laser illumination line is $1mm$. The distance between the illumination line and the camera pixel ranges from 0 to 15 mm. To make the diffusion approximation valid for the algorithm, we only use the images with the illumination to pixel distance Δy larger than 2 mm.

The simulated direct and global light images are shown in the first two rows in Figure 5.9. The global light image is the sum of the images captured with different Δy's except for $\Delta y = 0$. The inclusions can not be seen in the direct only image. For the global light image, because highly scattering property of skin, the contrast is low for some of the deeper inclusions, such as the solid circle in the right column. This makes the detection and localization for such objects (e.g. tumor beneath the skin surface) difficult. For each short-range indirect image, the image intensity is contributed in part by the indirect light that travels from the illumination line with a preset distance to the imaged scene point. As a result, compared with the global light image, the contrast of the inclusions are much higher for the short-range indirect images shown in the third and fourth rows of Figure 5.9. On the other hand, for larger pixel to illumination line distance, the SNR is low because there are less photons reaching the imaged scene point due to multiple scattering of light. In addition, because the non-zero support of the diffuse scattering kernel increases with the pixel to line illumination distance, the boundaries of the inclusions in the image becomes more blurry for larger distance. Despite low SNR and blurring in the short-range indirect images, using the proposed method, we are able to localize the 2D boundaries and estimate the relative depth for the inclusions.

For the input short range indirect images, the contrast of the image decreases with the inclusion depth, as shown in Figure 5.10 (a). This is because as the inclusions become deeper, most light reaching the pixel is contributed by the scattered light from the homogeneous medium without traveling through the inclusions. Another intuition is that the diffuse scattering phase function

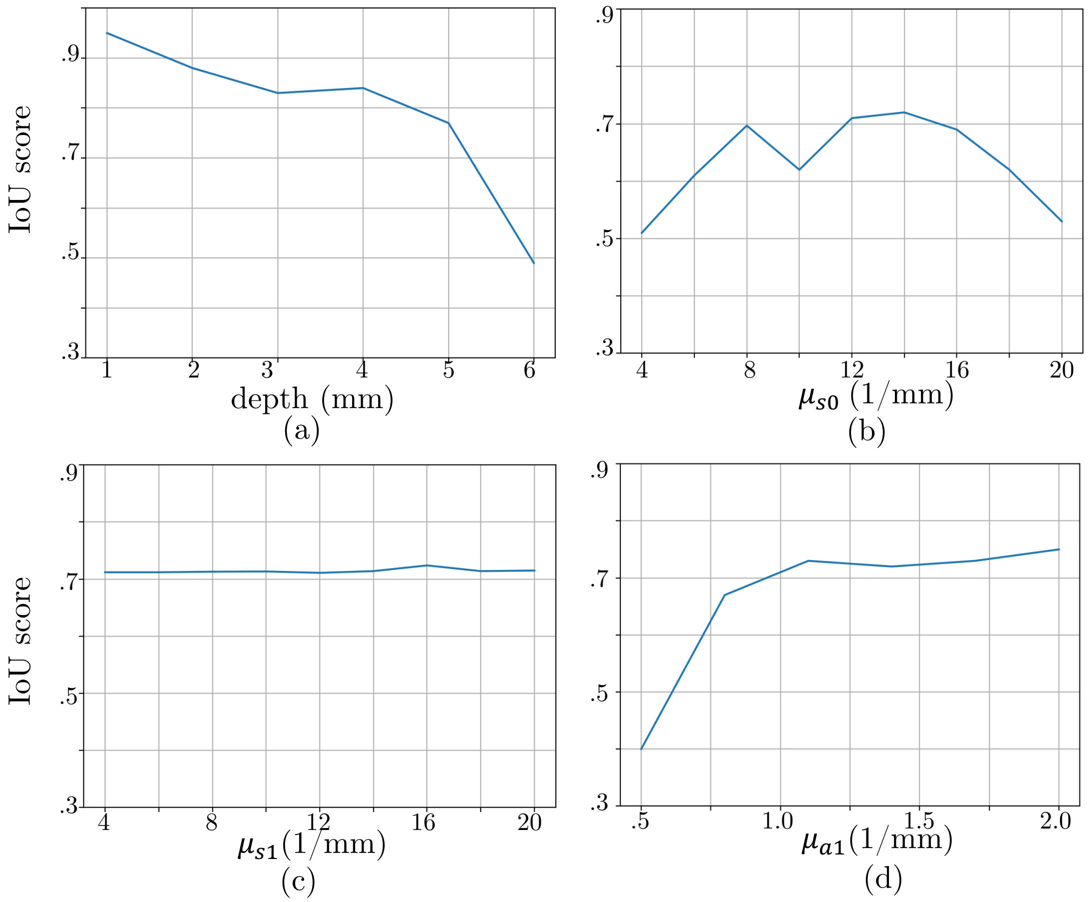


Figure 5.11: Performance with different (a) heterogeneity depth; (b) scattering coefficient μ_{s_0} of homogeneous background; (c) scattering coefficient of the heterogeneity μ_{s_1}; (d) absorption coefficient of the heterogeneity μ_{a_1}.

diminishes with the depth increase, as shown in Figure 5.4.

One key factor for the proposed method is the size of the diffuse scattering kernel. Smaller kernel enables faster optimization process, but it will lead to more errors in the convolutional approximation, hence less accurate results; while larger kernel leads to better performance, it induces more processing time. The choice for the size of diffuse scattering kernel is also related to the pixel to illumination line distance. In addition, as shown in Figure 5.4, the non-zero support region for the kernel varies with the pixel to illumination line distance. For large pixel to illumination line distance, the magnitude of the kernel would be small due to multiple scatterings, so the performance will saturate at certain distance. In Figure 5.10 (b), we show how the performance changes with the kernel size when the medium is human skin. The performance is evaluated using the IoU score of segmentation results for the inclusions. As we can see, for highly scattering medium like human skin, the performance saturates when the kernel size approaches 20 mm in diameter.

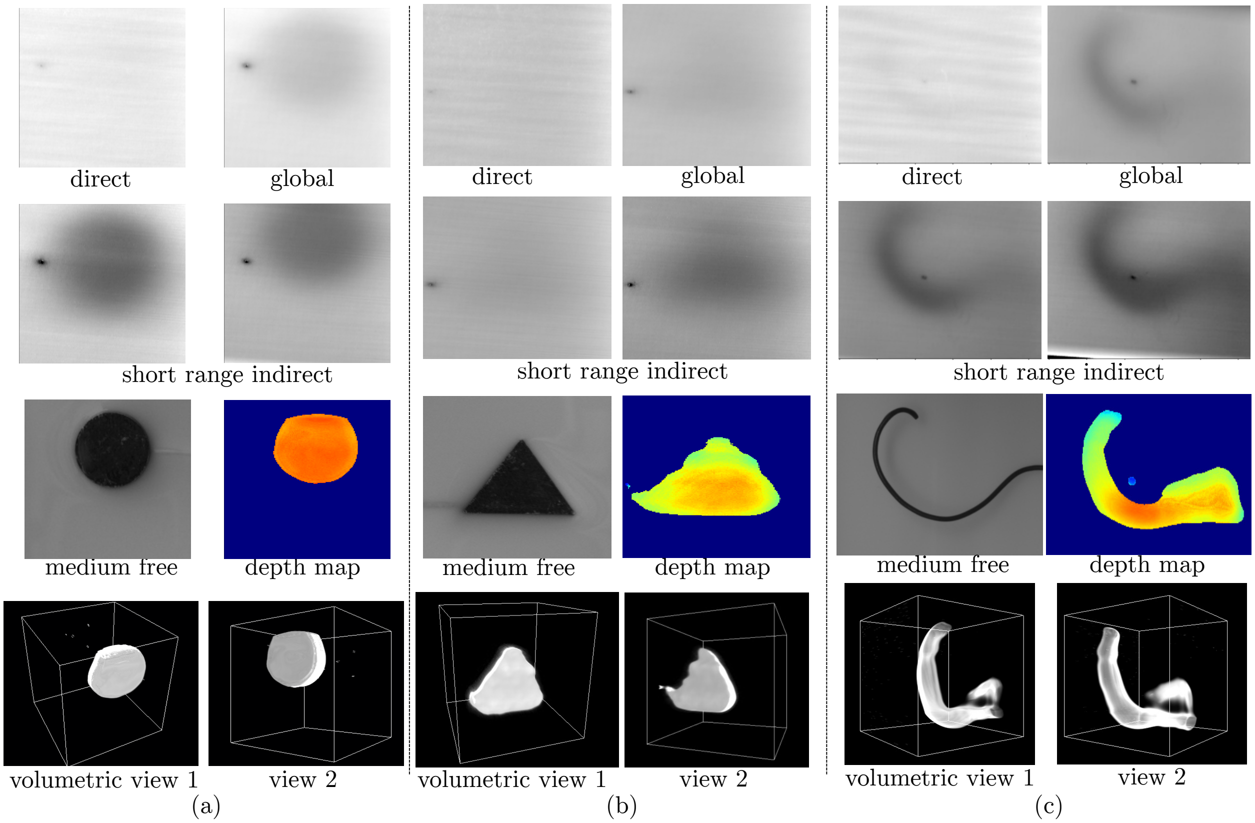


Figure 5.12: Real data images and results for single inclusion. The scattering medium is skim milk with no or little water dilution. Rows from top to bottom: the direct and global light images, the short-range indirect images with different Δy settings, the medium free image and masked depth map, two views of the 3D tomography results. (a) Solid circle plate inclusion. The inclusion boundary in the global and short-range indirect images are either blurred or shifted due to light scattering. The boundary is faithfully recovered from our method. (b) Solid triangle plate inclusion. The inclusion is either blurry or barely visible in the global light and short-range indirect images since the inclusion is relatively deep in the scattering medium. Our method is able to reconstruct the triangle structure despite highly scattering effects. (c) Curved thin black wire. The 3D wire structure is recovered in the 3D tomography results, even though it is not obvious at all in the short range indirect images.

Performance for different scene settings

The derivation of the forward model in Section. 5.3 is based on two assumptions about the scattering mediums: (1) the scattering coefficient of the surrounding homogeneous medium is large such that the light propagation direction distribution is isotropic; (2) the absorption coefficient discrepancy dominates the scattering property difference between the heterogeneous embedding and the surrounding medium. We evaluate the robustness of our method against the failure of those assumptions.

To this end, we perform four simulation experiments and evaluate the performance with different scattering property and scene settings. For all experiments, the measured images are rendered using Monte Carlo simulations. The performance is evaluated in terms of the IoU scores. We use the same denotation as in Section 5.3: μ_{s_0}, μ_{a_0} are the scattering and absorption coefficients for the surrounding homogeneous medium; μ_{s_1}, μ_{a_1} are the scattering and absorption coefficients for the embedded heterogeneous material. The performance is shown in Figure 3.4.

Performance vs. heterogeneity depth As seen, the performance of our method decreases with the depth of the inclusion since the image contrast reduces with depth. For small depth, although single-scattering events can dominate, the large image contrast of the heterogeneous medium makes the reconstruction task easier for the proposed method.

Performance vs. scattering coefficients of the homogeneous background An interesting observation is the parabolic-type performance curve. For lower scattering coefficient μ_{s_0}, the diffusion approximation starts to become less valid, resulting in modeling error. If the scattering coefficient μ_{s_0} is larger, multiple scattering events govern the photon propagation inside tissue, increasing the accuracy of our forward model. However, it becomes progressively difficult to recover the position of the heterogeneous object since now few photons actually sample the heterogeneity embedded at a particular depth and get detected by the detector.

Performance vs. scattering coefficients of the heterogeneity This set of experiments addresses the robustness with different scattering properties of the heterogeneous medium μ_{s_1}. The scattering coefficient of the homogeneous medium μ_{s_0} was kept constant. We noted that even though we vary μ_{s_1}, the performance of our method does not change much. The invariance of the performance is due to the sparse nature of the heterogeneous object inside the medium.

Performance vs. absorption coefficients of the heterogeneity For smaller μ_{a_1} values, the performance is lower due to the fact that the contrast of the heterogeneous object compared to the background medium is lower in the short-range indirect images. Though we assume that the change in absorption coefficient of the object is small compared to the background medium, the increase in contrast helps our algorithm to recover the location of the object.

Comparison with traditional DOT method

We compared our method with the traditional DOT method [17] quantitatively in terms of computational time and performance through simulation. The scene setup is a homogeneous medium with 3 rods embedded at certain depth with different absorption coefficient than the background medium.

For traditional DOT, We considered a fixed number of source-detectors (80 sources and 80 detectors) to reconstruct a volume of $64 \times 64 \times 8$ resolution. The reconstruction process took 15 minutes. The reconstructed volume was upsampled to the scene resolution and the IoU was computed to be only 0.18. For our method, we reconstructed the same scene using our setup with a resolution of $256 \times 256 \times 64$. The IoU from our method was 0.71 and it took 4 minutes. The results show that we are able to perform reconstruction of much higher resolution and accuracy using our method compared to traditional DOT.

5.6.2 Real Data

We test the proposed method on images captured using the calibrated imaging setup shown in Figure 5.7 (a). We choose the embedding medium to be milk with little or no water dilution because its scattering property is close to human skin and it can be well described using the diffusion approximation. Because the small FOV and high camera resolution, the laser line illumination spans multiple pixels in the image. We calibrate the laser light source for the its intensity and the width of laser beam. More specifically, we use a ideal white diffuser with albedo close to one, and illuminate the diffuser with the laser line illumination. Then we extract the 1D profile for the laser line illumination, by averaging the intensity along the reflected illumination line. During the optimization in Equation 5.17, the rendered image is convolved with the measured 1D laser profile to account for the width of the line illumination.

During imaging, we use the short range indirect images with pixel to illumination distance Δy ranges from $20mm$ to $40mm$, such that the light source to sensor distance is large enough for the diffusion approximation. This configuration is different from simulation because the laser beam spans more pixels in the real data. We capture the HDR images to include the large range of image intensity under laser illumination. During optimization, for efficiency, we set the size of the diffuse scattering kernel to be $30mm$. The initialization of the reconstructed volume for all experiments is set to zeros. The measured 1D laser profile is convolved with the rendered images to account for the laser span of multiple pixels. For each scene, we manually select a homogeneous region and fit the scattering properties using the dipole model in Equation 5.2. For all the results, we use 300 iterations and it takes around 5 minutes for optimization on a workstation with TitanV GPU.

We test on scenes with single and multiple inclusions within the scattering medium. In Figure 5.12, we show the captured images and reconstructions for single inclusion. Note that in Figure 5.12 (a) and (c), the inclusion boundaries in the global and short-range indirect images are blurred due to multiple light scattering. Compared with the short range indirect images, the contrast of the inclusions is lower in the global image. In addition, as shown in Figure 5.12 (b), the inclusion is barely visible if it is deep below the surface. Our method is able to localize the boundary and reconstruct the 3D structures (e.g. the wire structure) despite low visibility and lack of contrast in the input images. Similarly, as shown in Figure 5.13, for multiple inclusions, the boundary of the inclusions and their relative depths can be recovered although the contrast and visibility of the inclusions in the input short-range indirect images are low due to highly light scattering. The inclusions are up to 8 mm beneath the whole milk surface and no water dilution is added. The dark dots in the images are mask of the light reflection from the protection glass surface for the MEMS mirror, which cannot be controlled and can only be removed in the clean room to prevent the mirror from being contaminated.

5.7 Limitations

In this work, we have assumed that the light scattering in the medium is dominated by the high-order scattering events such that the radiance can be modeled using the diffusion equation. However, for less dense medium or heterogeneities that are close to the surface, the single scattering

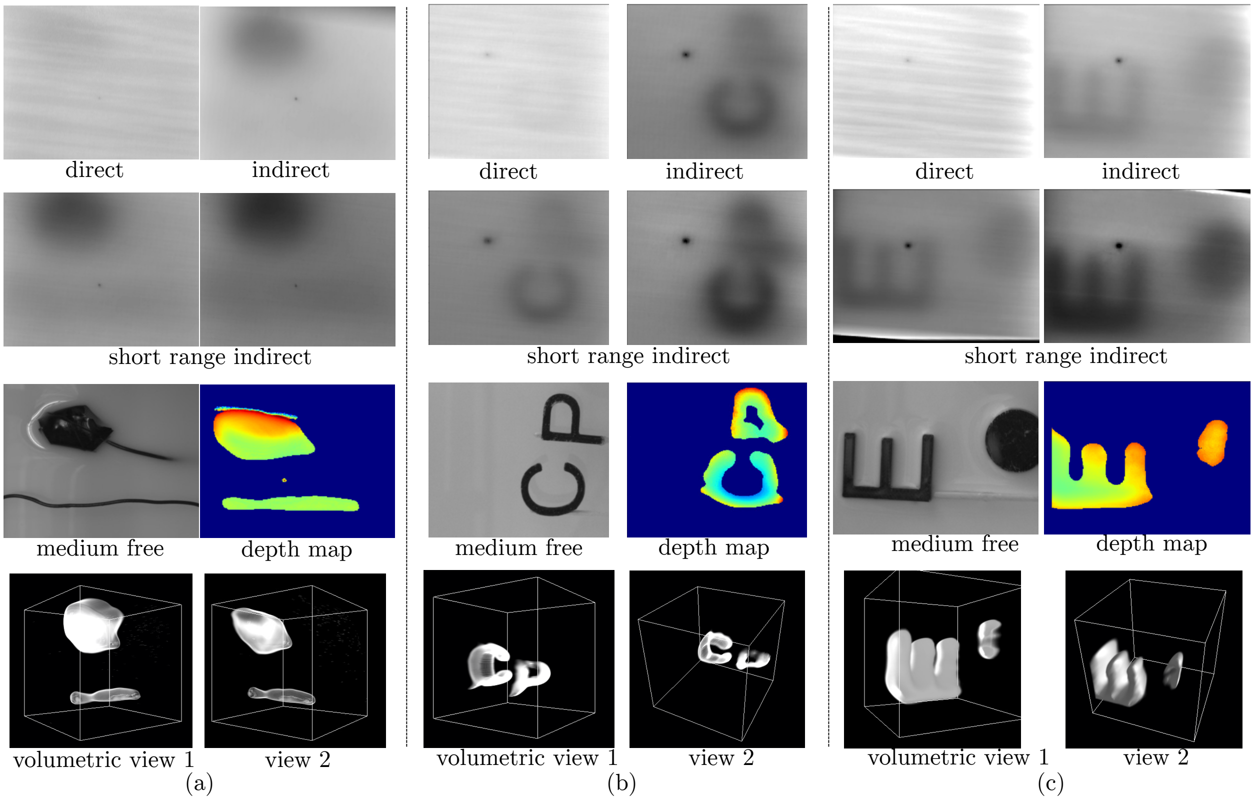


Figure 5.13: Real data images and results for multiple objects inclusions. The scattering medium is skim milk with no or little water dilution. Real data results. Rows from top to bottom: the direct and global light only images, the short-range indirect images with different Δy settings, the medium free image and masked depth map, two views of the 3D tomography results. (a) Thin wire and black tape blob. The thin wire is barely visible in the input images due to its small width and light scattering of the medium. However, the location and boundary can be recovered using our method. In (b) and (c), the 3D structures are recovered even though the letters are blurred in the input short range indirect images because of light scattering. In (c), the inclusions are up to 8 mm beneath the whole milk surface and no water dilution is added.

events becomes more evident. As a result, our method cannot handle well the less dense medium (*e.g.*severely diluted milk) or heterogeneities very close to the surface ($\leq 1mm$ beneath human skin through simulation). To handle theses cases, we need to include the single scattering component into the forward model.

5.8 Conclusion and Future Work

Our work addresses two fundamental limitations of existing diffuse optical tomography methods: low resolution reconstruction and high computational complexity. We overcome these limitations by (1) extending the design for short-range indirect subsurface imaging to a verged scanning projector and camera configuration and (2) a novel convolution based model and efficient computational algorithm for estimating the subsurface medium with heterogeneous structures. This allowed us to recover detailed heterogeneous structures immersed up to $8mm$ deep in a highly scattering medium, such as whole milk, for the first time. Avenues of future work include using other source spectra (near-infra red) to recover structures deeper within tissue, and using resonant MEMS scanning for capturing subsurface videos of dynamic structures, such as blood flow in microscopic capillary veins.

Chapter 6

Real-time Visual Analysis of Microvascular Blood Flow

6.1 Introduction

Microcirculation takes place in part of the circulatory system embedded in tissue that involves the smallest vessels and where diffusion of nutrients and oxygen into the cells and removal of CO_2 and waste from the cells take place. Monitoring of microcirculation is useful for diagnosing of vascular conditions and in monitoring patients for cardio-respiratory insufficiency.

Sidestream Dark Field (SDF) [69] video imaging was developed as a non-invasive imaging approach for real-time visualization of superficial microvascular flow. However, analysis of these videos is currently limited by manual or semi-manual operation and coarse sampling techniques, which makes quantitative analysis of microcirculatory status and response to disease and treatment difficult and subjective [71]. We aim to remedy that. One of the portable SDF imaging devices is shown in Fig. 6.1(a). As depicted in Fig. 6.1(c), illumination is provided by the green light-emitting diodes (LEDs) arranged in a ring formation. The wavelength (λ=530 nm) of the illumination is chosen to maximize light absorption by the red blood cells (RBCs). The tissue embedding the capillaries scatters and reflects the illumination back to the camera, making the capillaries imaged as dark curvilinear structures against the brighter background. The LEDs and the lens system are optically isolated to prevent the illumination generated by the LEDs from contaminating the images.

Despite that the design is optimized for microcirculatory imaging, as shown in Fig. 6.1(b), it is not easy to extract physiological features from SDF videos, such as the blood flow velocity, for several reasons: (1) Subsurface scattering: scattering of light on the path from the capillaries to the camera increase observed intensity of the vessels, reducing contrast of the images; (2) Defocus: capillaries are embedded at varied depths within the tissue while the depth of field of the camera is fixed to obtain desired magnification. So some capillaries in the field of view appear blurred, making their features more difficult to estimate; (3) Sensor noise that further reduces quality of images; (4) Limited texture: low diameter capillaries of interest comprise only a small part of the image, most of it is occupied by tissue without substantial texture and in addition, plasma in the

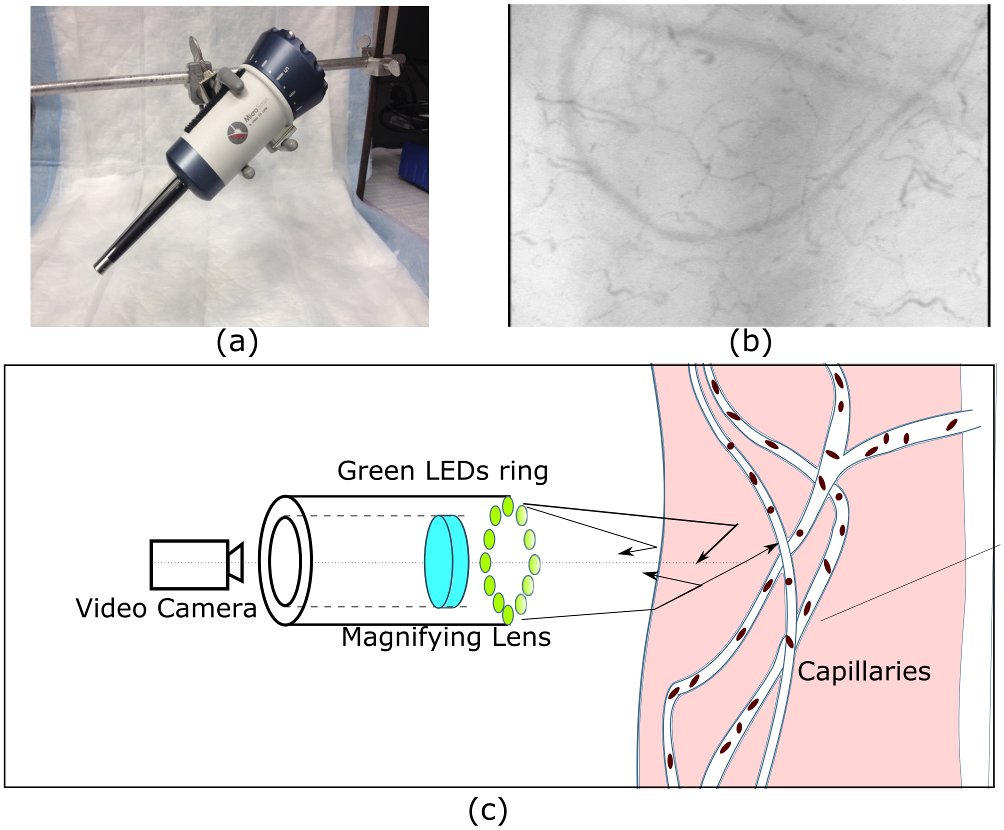


Figure 6.1: Sidestream Dark Field Imaging[69]. (a) Portable SDF imaging device used for microcirculatory monitoring. (b) One frame of the microcirculatory video. (c) The LEDs, arranged and optically isolated around the lens system, emit light optimized for red blood cell absorption. Due to defocus, subsurface scattering of light, sensor noise, sensor drifting and limited texture of the tissue, it is not easy to extract physiological features from the SDF video.

capillaries is transparent, reducing texture in the frames even further, so traditional texture-based image feature extraction methods will likely fail; (5) Sensor drift during video capture: field of view changes due to the motions induced by heart beat and respiration of the subject and movement of the device itself, relative to the observed tissue.

In this chapter, we present an end-to-end, automated framework for real-time analysis of microcirculation including vessel detection, heart rate, breathing rate, blood flow velocity estimation as well as variations of flow distributions over time during bleeding and resuscitation stages. Our work can enable new research in critical care, helping correlate heart rate and breathing cycle with flow distributions and studying effects of interventions and protocols in real-time for bed-side patient care. In comparison, most previous works either included significant manual interactions and were not real-time, or are tailored to high quality 2D images or 3D volumes that do not work for SDF videos.

The underlying principle of our approach is that diagnostically useful information must be extracted quickly, enabling the user to make determinations about microcircluatory flow in real time, rather than off line as is done currently, and ultimately enable making clinical decisions instantly at the bedside. To this end, we present a framework consisting of multiple stages including video stabilization, enhancement, micro-vessel extraction and automatic estimation of the micro blood flow statistics from SDF videos.

Our method has been used in a critical care experiment conducted carefully to analyze the microcirculatory blood flow of subjects in different health conditions. In the experiment, healthy pigs have been anesthetized and subjected to induced slow bleeding (20 ml/min) for about 2 hours. Then the subjects were fluid resuscitated to expand the plasma volume. Microcirculatory videos were captured at different stages of the experiment to monitor changes in the micro blood flow. 96 videos of 18 pigs were collected using a SDF imaging device for each bleeding/resuscitation stage. Our method was then applied to extract physiological information from the videos. As a result, the extracted informative microcirculatory features form distributions that are consistent with the intuition of expert clinicians.

6.2 Related work

Image based microcirculatory blood flow measurements have been studied using Laser speckles [19, 32]. More recently, skin perfusion measurement based on laser speckle was proposed in [157]. Instead of images or videos of the microcirculatory blood flow, these methods leverage complex speckle patterns. In Sidestream Dark Field imaging system [69], microcirculatory blood flow is analyzed while the labeling of capillaries is done manually [43].

The vessels in the image are often detected as centreline structures [103, 125, 170, 174, 192] either by using filters [103, 192], intensity profiles [125, 174], or trained regressors [170]. Then, level-set methods are used to locate the centreline more precisely [79, 195].

In [177], various optical flow approaches are studied. It was shown that by using an objective with a non-local term, the classical optical flow formulations can achieve competitive results. For motions of deformable objects, the motion estimation problem is often formulated as optimization solved by inverse compositional image alignment [121], supervised-learning of descent direction

[185], and data-driven descent [185]. In our case, with high level of noise, highly deformable blood flow patterns, and small dimensions of capillaries, it is very difficult to track the flow on a frame-by-frame basis.

To get motions that are more obvious and easier to detect, video motion magnification method has been proposed in [207]. Extensions have been put forward to either reduce the noise in the motion magnified video [207] or achieve real-time running speed [198]. Because of the high level noise in the SDF videos, applying any of those methods directly would likely amplify the noise as well.

6.3 Micro-vessel Extraction from Video

The contrast of the SDF images is greatly reduced by the presence of the subsurface scattering and sensor noise. This makes it difficult to detect the capillaries from any single frame in the video. One option is to detect the capillaries from the minimal image, where the values of the pixels are set to the minimal intensity across frames at that pixel location. However, the input videos are not stable because of motions introduced by heart beat, respiration, and sensor position drift. So we need to stabilize the video before extracting vessel skeletons.

6.3.1 Video Stabilization

After motion due to heartbeat, breathing and sensor position drift is eliminated, the stabilized video will mainly consist of the blood flow in the capillaries. For efficiency considerations, we base video stabilization on motions of the patches that are corresponded between frames using template matching. Since the microcirculatory videos are captured carefully to avoid unnecessary motion of sensor relative to subject, frame-to-frame changes are limited and smooth. Thus the correspondence between patches in different frames can be estimate. In addition, patch-based stabilization method enables including variations of the patch motions in a frame introduced by deformable properties of the tissue.

Because the videos are effectively textureless in most parts of the frames, we need to select the optimal patches for finding correspondence in the stabilization process. In our method, we select the patches in which the variance of intensities is above a pre-set threshold such that the selected patches include enough texture for matching.

6.3.2 Vessel skeleton extraction

After stabilization, we have registered frames from which the skeletons of vessels can be extracted. However, as shown in the first column of Fig. 6.2, due to subsurface scattering and imaging noise, the contrast in individual frames is too low for extracting vessel segments. Even worse, the transparent plasma travelling through the capillaries may make vessels invisible in some segments of a frame. So we first need to produce a vessel-enhanced image. Based on the fact that the capillaries with red blood cells are usually darker in the frames, we can take the minimal value of each pixel across all the frames to achieve that goal. This method works under assumption that for every pixel of the vessel there is at least one frame in which a red blood cell passes through it. This assumption is true for most cases since the duration of the microcirculatory videos (20 seconds) is long enough for the red blood cells to pass through all the active vessels in the frame.

Then the vessel enhanced image is denoised by applying anisotropic diffusion filtering. It not only reduces the imaging noise while leaving the edges in the vessel enhanced image unharmed, but it also smoothes the parts of the image along the structures between the edges. This results in vessel segments with a smooth appearance so they can be detected more easily. The filtered vessel enhanced images are shown in the second column of Fig. 6.2.

To detect the vessel skeletons, we first estimate the Hessian matrix for each pixel in the vessel enhanced image. Then the profile for each pixel is extracted along the direction of the eigenvector of Hessian corresponding to the largest absolute eigenvalue. The pixel will be selected as a vessel skeleton pixel if the profile has a groove in the middle and increases towards both sides of the groove. To find the vessel skeletons with such profile, we use the method proposed in [174] that was designed to find the centreline of curvilinear structures.

Let $\mathbf{n} = (n_x, n_y)$ with unit length be the direction in the eigenvector of the Hessian Matrix H corresponding the largest eigenvalue. The second-order Taylor expansion of pixel at $\mathbf{x}$ along $\mathbf{n}$ is given by:

$$p(t) = r + r_n t + \frac{1}{2} r_{nn} t^2 \tag{6.1}$$

where $p(t)$ is the pixel intensity at the position $\mathbf{x} + t\mathbf{n}$; r, r_n and r_{nn} are the pixel intensity at $\mathbf{x}$, the first-order derivative of the intensity in the direction $\mathbf{n}$ and the second-order derivative of the intensity in the direction $\mathbf{n}$ respectively. For a profile across the vessel, The center of the groove is located at the zero crossing of the first derivative of the profile:

$$t = -\frac{r_n}{r_{nn}} = -\frac{\nabla \mathbf{r}^T \mathbf{n}}{\mathbf{n}^T H \mathbf{n}} \tag{6.2}$$

where $\nabla \mathbf{r}$ is the gradient of the image at $\mathbf{x}$. In the image coordinate, the offset of the zero-cross from $\mathbf{x}$ is $(p_x, p_y) = (tn_x, tn_y)$, with t estimated in Equation.6.2. The pixel $\mathbf{x}$ is on the vessel skeleton if $|p_x| \leq \frac{1}{2}$ and $|p_y| \leq \frac{1}{2}$. To eliminate the falsely detected vessels introduced by imaging noise, we use the maximum eigenvalue of the Hessian matrix to select the detected vessel skeletons.

The example results of the skeleton extraction are shown in Fig. 6.2. By comparing with the vessels manually labeled by human experts , we find that the vessel skeleton extraction method is able to locate most of the vessels in the frame. Although there is a potential for a few missing and false detections, the main objective of our work to extract informative statistics of the physiological importance, and not the analysis of the individual vessels - should not suffer much. Hence, the obtained skeletons can be used as reliable inputs to the subsequent processing steps.

6.3.3 Comparisons

We have compared the performance of our methods with other vessel extraction methods. For vessel skeleton extraction, our method yields a recall:87.90 % and false alarm rate:0.65%. In comparison, EF filters [58] on the minimal frame followed by adaptive thresholding on the filter

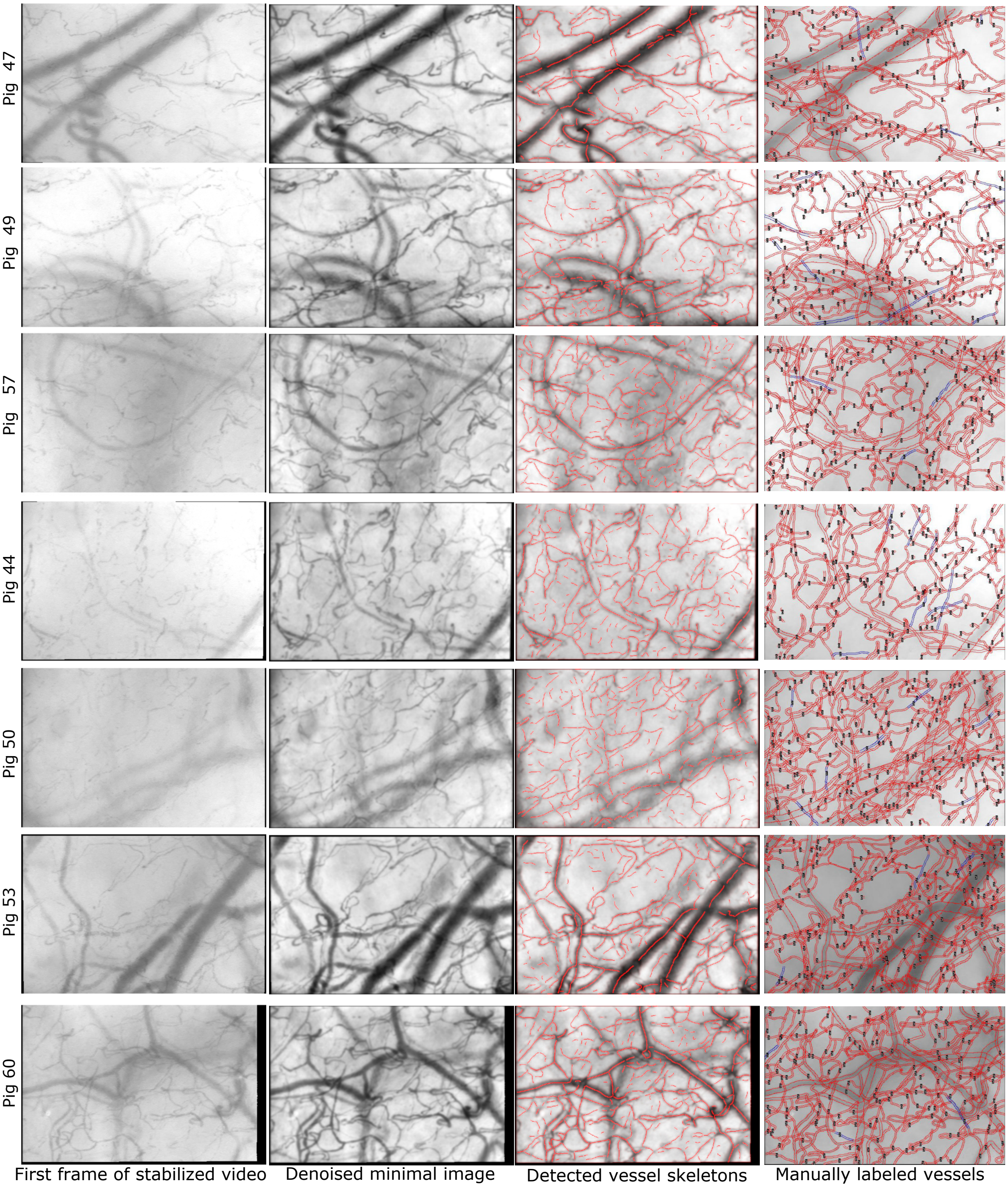


Figure 6.2: The vessel skeletons are extracted from the minimal image across the frames. First column: the first frames of the videos. Due to subsurface scattering and transparency of plasma, it is hard to detect capillaries from a single frame. Second column: denoised minimal image across all the N frames. In our case $N = 200$. Third column: extracted vessel skeletons. Fourth column: manually painted vessels. The index and status of subjects in each row: Pig 44, before resuscitation; Pig 50, before resuscitation; Pig 53, end of baseline; Pig 60, end of baseline.

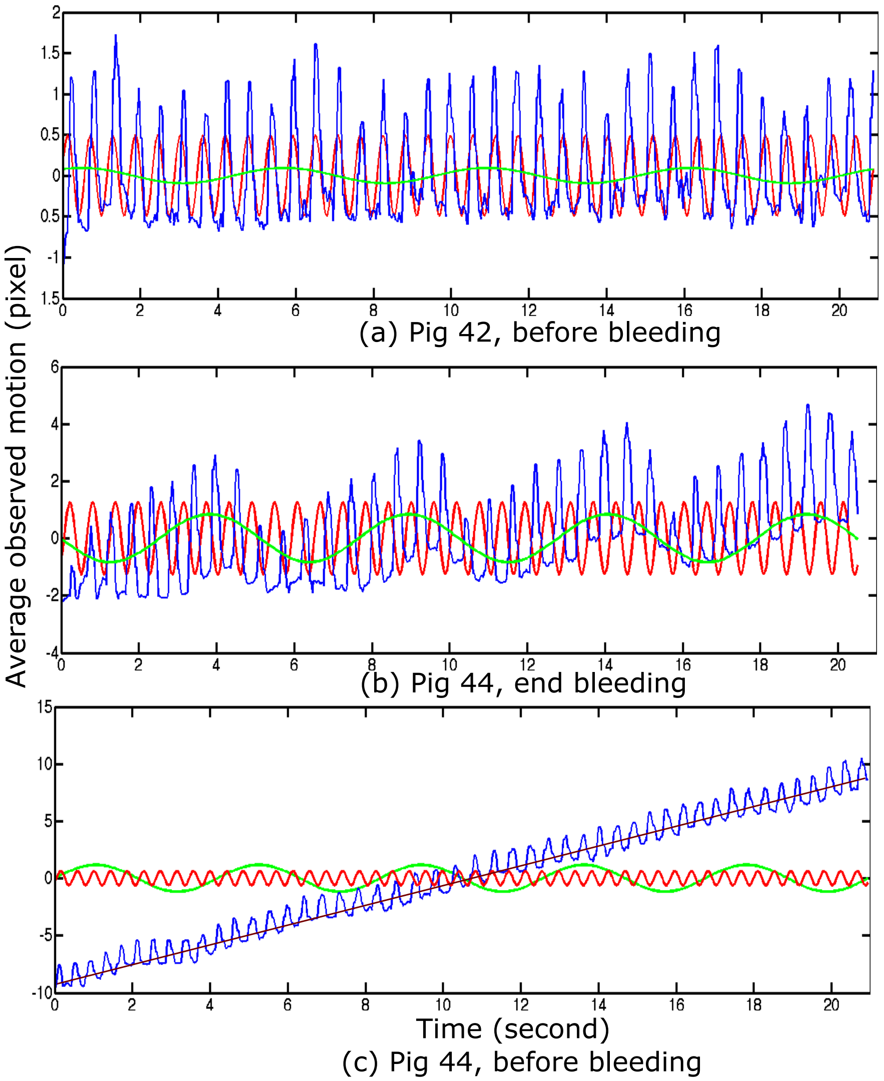
response yields 51.86% and 0.36% in accuracy and false alarm rate. 2D OOF filters [103] on the minimal frame followed by a adaptive thresholding on the filter response yields 70.79% and 1.29%. Filter learning plus tree regression [170] on the minimal frame yields 27.81% and 17.55%. Finally, filter learning plus tree regression on the video yields 27.47% and 14.91%. Note that these comparisons are much worse if applied to original videos without applying the sub-surface scattering reduction method.

6.4 Physiological Measurement from Video

Heartbeat and respiration rates can be obtained as side products of the video stabilization process. Those physiological measurements can be used along with the microcirculatory blood flow parameters, to further aid diagnosis and monitoring processes. As the observed motion introduced by the heart beat and breathing also depends on the location where the microcirculatory videos are taken, the measured motion can be used as a guidance for the clinician to determine the location of target tissue considered for diagnosis. In addition, although in clinical practice the assessment of the heart rate and the respiratory rate already exist via dedicated, specialized monitors, it is not known whether and how their variations impact physiology of tissue blood flow. The measurements of these signals thus provides an opportunity to study these interactions in a live subject concurrently with flow information, and generate further knowledge in the field.

We decompose the observed cross-frame motion into heartbeat and respiration motions based on their frequencies. More specifically, the respiratory is the motion component in the $[.1\ , .5]$ Hz frequency range in the Fourier transform of the averaged observed motions of patches in the unstabilized video; and the heartbeat is the motion component in the $[.5\ , 5]$ Hz frequency range. In the corresponding frequency ranges, the frequencies of the heartbeat and respiration motions are determined as the frequencies where the local maxima of magnitude in the Fourier domain occur. The magnitudes depend on the status of the subject and the location where the video is taken. For Pig 42, as shown in Fig. 6.3(a), most of the observed motion is due to the heartbeats. For Pig 44 at the end of bleeding, both the respiration and heartbeat motions are more significant. For Pig 44 before bleeding, the sensor drifting dominates the observed motion, while the other two components can still be reliably identified. This last observation has important practical implication, since apparently the perfect stabilization of the sensor probe against the subject tissue is not necessary for extracting reliable physiological information from SDF imaging videos.

After the videos are stabilized, we estimate local blood flow velocity using skeletons to identify individual vessels. Even though we have now vessel-enhanced images with improved contrast, it is still difficult to determine blood flow from video, because signal to noise ratios are still low with the effects of subsurface scattering and high imaging noise. To make the blood flow more detectable, we use the motion magnification method proposed in [207]. In general, motion magnification is achieved by amplifying the frequency components within a given range for each voxel in the video. This method is based on the intuition that for one point in the video with repeating motion, the frequency of change in the intensity depends on the speed of motion that passes through that point. Motion is then amplified by magnifying the frequency component corresponding to it. The first frames in the original video and the motion magnified video are shown in Fig. 6.4. Note that



(a) Pig 42, before bleeding

(b) Pig 44, end bleeding

(c) Pig 44, before bleeding

Figure 6.3: The averaged observed motions (blue) across the frame and their components. Motion components due to heartbeat and breathing are colored in red and green respectively. (a) Pig 42 at the end of baseline (before bleeding). Most of the observed motion is due to the heart beat. (b) Pig 44, end of bleeding. Both heartbeat induced motion and breathing motion are obvious. (c) Pig 44, baseline. The sensor position drift (shown in brown) dominates the averaged observed motion, but the physiologic components can still be clearly identified.

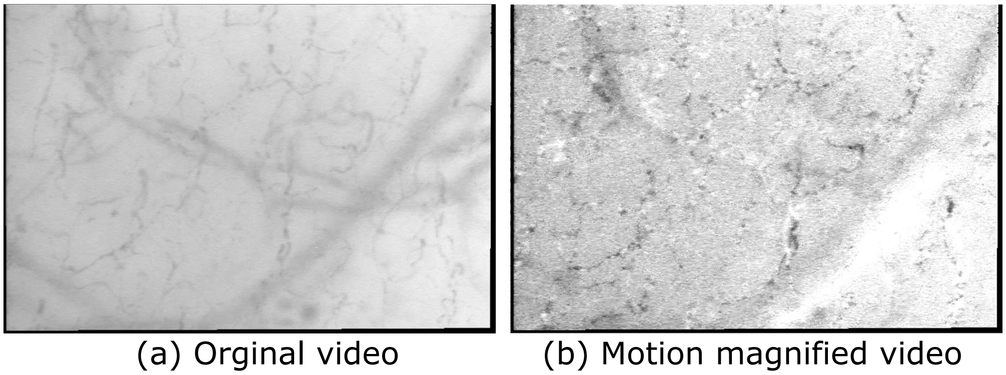


Figure 6.4: The first frame in the original video and the motion-magnified video. Noise in the original video is magnified along with the blood flow motion.

the contrast between the plasma and red blood cells is enhanced in the motion magnified video. On the other hand, background noise has also been magnified since the frequency ranges of noise and blood flow overlap.

Blood flow velocity is estimated from the motion magnified video and the vessel skeletons. There is significant variation in flow velocity across vessels of different shapes and sizes, as well as due to physiological variations. This makes the optical flow method hard to work well in our case. One example is shown in Fig. 6.5. For a motion magnified video, the color coded optical flows estimated using the method in [177] for two consecutive frames are shown in Fig. 6.5(c) and Fig. 6.5(d). The corresponding vessel enhanced image for the video is shown in Fig. 6.5(a). Due to the imaging noise and difficulty in tracking the flow, the estimated optical flows for the two consecutive frames are very inconsistent, making the optical flow estimation unreliable.

Since the diameters of capillaries in the microcirculatory videos are small, the blood flow motion in the video can be reliably approximated by 1D motion along the vessel skeletons. With this approximation, blood flow velocity is estimated from the Epipolar-Plane Image (EPI) along the vessel skeleton length. More specifically, as the blood flow speed along a vessel is relatively constant, its EPI image will have stripe patterns in it. The blood flow velocity for a vessel segment is then estimated from the orientation of the EPI image stripe pattern. Based on the rotation property of the Fourier transform, the rotation of a function by an angle in the image domain will yield a rotation in the Fourier domain with the same angle. So the slope of the stripe pattern in the EPI image corresponds to the dominant orientation in its Fourier transform. Thus we can estimate the velocity of the blood flow by finding the dominant orientation in the Fourier domain of the EPI image. This dominant orientation of the Fourier transform can be found by fitting the line passing through the origin, such that the second-moment inertia is minimized. We can solve this via a standard inertia minimization process.

The fitted line with the optimal angle indicating the dominant orientation in the Fourier domain is plotted in Fig. 6.6(d) in green. According to the rotation property of the Fourier Transform, the dominant orientation in the EPI image is the estimated orientation in the Fourier domain rotated clockwise by 90 degrees, as shown in Fig. 6.6(b). Although there are various orientations of patterns in the EPI image because of temporal and spatial fluctuations of blood flow velocity,

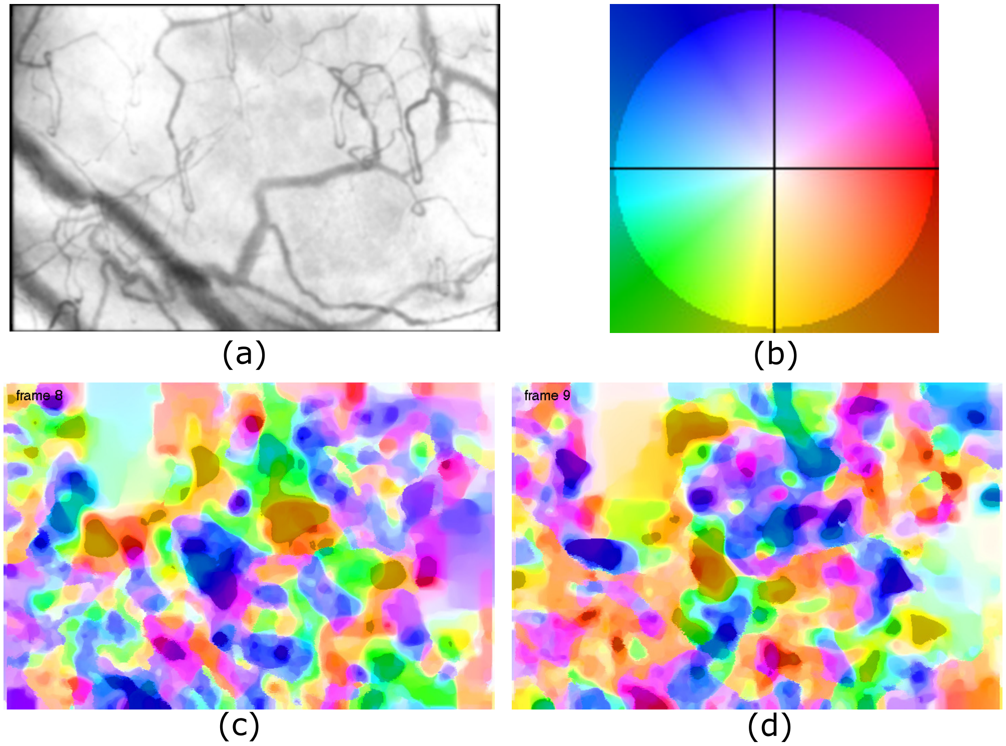


Figure 6.5: The blood flow speed estimation using optical flow method in [177]. (a) The vessel enhanced image. (b) The color wheel showing colors corresponding to directions and magnitudes of optical flow. (c) The color coded optical flow for frame 8. (d) The color coded optical flow for frame 9.

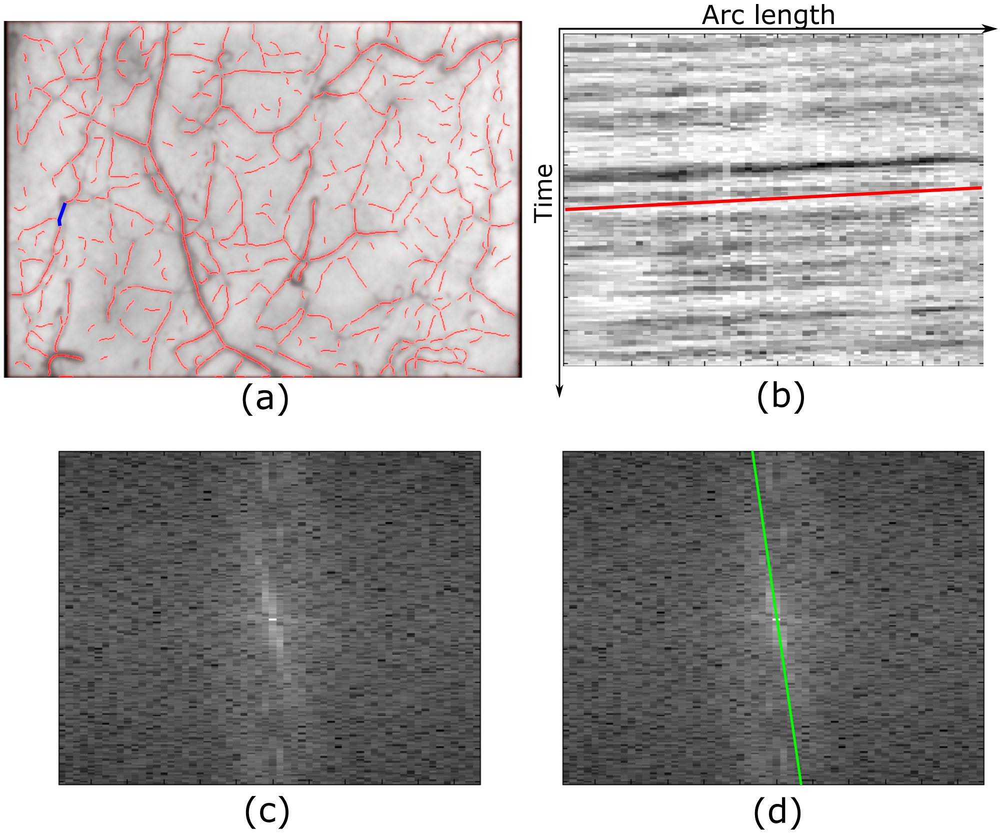


Figure 6.6: Blood flow velocity estimation. (a) The extracted vessel skeletons. The vessel segment for which the flow velocity is estimated is colored in blue. (b) The EPI image of the blue colored vessel segment in (a). (c) The Fourier Transform of the EPI image. (d) The dominant orientation of the Fourier Transform is plotted as the green line. The corresponding line showing the dominant orientation in the EPI image is plotted in red in (b).

the proposed method is still able to extract the dominant orientation, and therefore estimate the velocity.

In our experiments, the proposed method fits the task better than the optical flow approach because it takes into account data from multiple consecutive frames, while the optical flow method usually takes into account only two consecutive frames. Instead of measuring the blood flow velocity in one specific frame, our method actually measures the average blood flow velocity from multiple frames, which is more robust to the noise in the videos. By varying the extent of averaging, we can control temporal resolution of blood flow velocity estimation. It must be however noted that the frame-to-frame noise in our videos limits in practice the minimal time scales of this estimation.

6.5 Critical Care Case Studies

In this section we will relate the estimated blood flow velocity distribution across all vessel segments detected in the field of view, to the status of the test subjects in the bleed and resuscitation phases of the experiments in order to evaluate consistency of our method with knowledge and intuition of expert clinicians.

The critical care experiment procedure is shown in Fig. 6.7. All experiments were performed in accordance with NIH guidelines under protocol approved by the Institutional Animal Care and Use Committee of the University of Pittsburgh. Three Yorkshire Durock pigs (average weight of 30.6 kg) were fasted overnight prior to the study. Anesthesia and the surgical preparation have been performed following procedures described in [71]. Briefly, following induction of general anesthesia and endotracheal intubation, arterial and central venous catheters were inserted and the animals allowed to stabilize for 30 minutes. During this time the SDF probe attached to a vise clamp was positioned in the pigs mouth under the lounge to visualize the sublingual microcirculation. Care was taken to obtain a long-term stable image with minimal pressure artifact and good visualization of the microcirculation as defined by the optimal focal length and illumination to visualize the largest number of capillaries within the viewing frame as previously recommended in [42]. At the end of the baseline period the initial video was collected (Baseline). All videos were 20 seconds in length at 10 frames per second. Then the pigs were bled form the arterial catheter at a fixed rate of 20 ml/min until the mean arterial pressure decreased to 30 mmHg. Once at this pressure, bleeding was stopped and a second video was captured (EndBleed). The subject was kept in this hypotensive state for 90 minutes with video images captured at 60 minutes into the hypotensive state (AfterBleed) and again at 90 minutes (BeforeResusc). Then the pigs were fluid resuscitated with Hextend (equal volume to shed blood) at 60 ml/min. At the end of this fluid resuscitation period another video was captured (EndHextend). Then the animal was further resuscitated in a protocolized fashion as previously described with more fluid if the cardiac output was less than baseline and norepinephrine if mean arterial pressure was less than baseline for an additional 120 minutes and a final video image was taken (AfterHextend). Since many animals became unstable before 90 minutes of hypotension or did not survive 120 minutes after the start of resuscitation, some animals did not have BeforeResusc and AfterHextend time point videos collected. So, a 20-0second microcirculation video clip was captured at each of the six stages described above: (1) Baseline: right before the bleeding; (2) EndBleed: at the end of bleeding; (3) AfterBleed: 60 min-

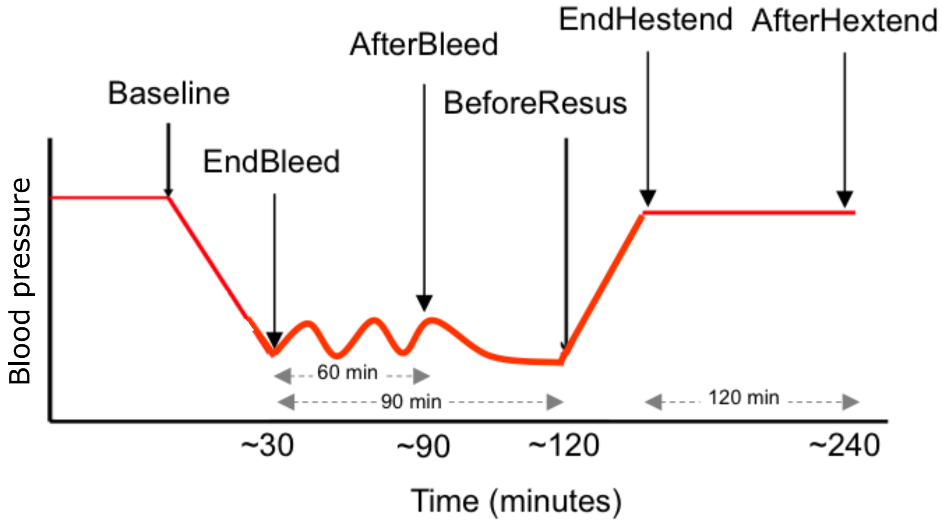


Figure 6.7: Setup of the experimental procedure. 18 pigs are observed carefully at various stages of bleeding and resuscitation.

utes after the end of bleeding; (4) Before resuscitation: 90 minutes after the end of bleeding; (5) End of resuscitation: the end of the resuscitation process, in which the Henxtend fluid is infused intravascularly; (6) After resuscitation: end of observation period.

From the point of view of current knowledge of physiology of the observed processes, as the blood pressure decreases due to bleeding, a general reduction in blood flow velocity is expected. It should be manifest by a shift of the distribution of velocities across vessels towards lower values of velocity. Although resuscitation should intuitively led to an increase of microcirculatory flow, the temporal relation between restoration of arterial pressure and cardiac output to microcirculatory flow is complex and not yet fully understood. Still, one would expect that if resuscitation efforts were successful, that microcirculatory blood flow would return to baseline values.

The estimated distributions of the blood flow velocity estimated from the corresponding videos are consistent with the above intuition. The results for five pigs are shown in Fig. 6.8. For better visualization, in the right column we show only the results for three most important stages in the process. For Pig 44 and Pig 60, the blood flow in the capillaries diminishes after bleeding as the blood pressure and the vitality of the pig deteriorate. This change has been reflected in the figure as the flow velocity distribution, shown in the green curve, squeezes towards a lower values. In addition, the number of capillaries with slow flow velocity decreased after resuscitation as compared to the after bleed phase. This is consistent with physiologic expectations, and represents the opening of capillary beds that were previously closed probably due to insufficient input pressure during shock. Given that this protocol was intended to study the individual responses of each animal to hemorrhage, blood volume shed was different between animals. Pig 44 and 60 had 534 ml and 760 ml, respectively, which represented 23% and 36.7% of their calculated total blood volume, respectively. This analysis also demonstrated how for example pig 44 had a lower relative increase in capillaries with slow flow, than pig 60, which is consistent with having had a less intense response, to a less intense injury.

For Pig 47, the difference in the blood flow velocity distributions between the baseline and after bleed is smaller than those for Pig 44 and Pig 60. This is because more probe pressure on the tissue was introduced during capturing the microcirculatory video for Pig 47 in the baseline stage, making the blood flow suppressed at that stage. Our method has reflected such measuring artifacts during capture. For Pig 55, there are only 5 stages in total since the pig died before last stage.

6.6 Determine Local Flow Velocity and Type

The analysis of local blood flow motion patterns serves as another aspect to measure the response of the micro-circulation system to the hemorrhage and resuscitation processes. For example, the cardiorespiratitory insufficiency caused by blood pressure loss is usually spatially variant. Such spatial variance demonstrates how the local micro-circulation system reacts to the blood pressure reduction. To visualize the spatial variance in the change of blood flow, we have designed the motion features to represent the local flows motions based on 3D convolution with pre-defined spatial-temporal filters. On top of that, a cascade of classifiers are trained to distinguish between different flow types, enabling us to localize the abnormal flows due to the loss of blood pressure.

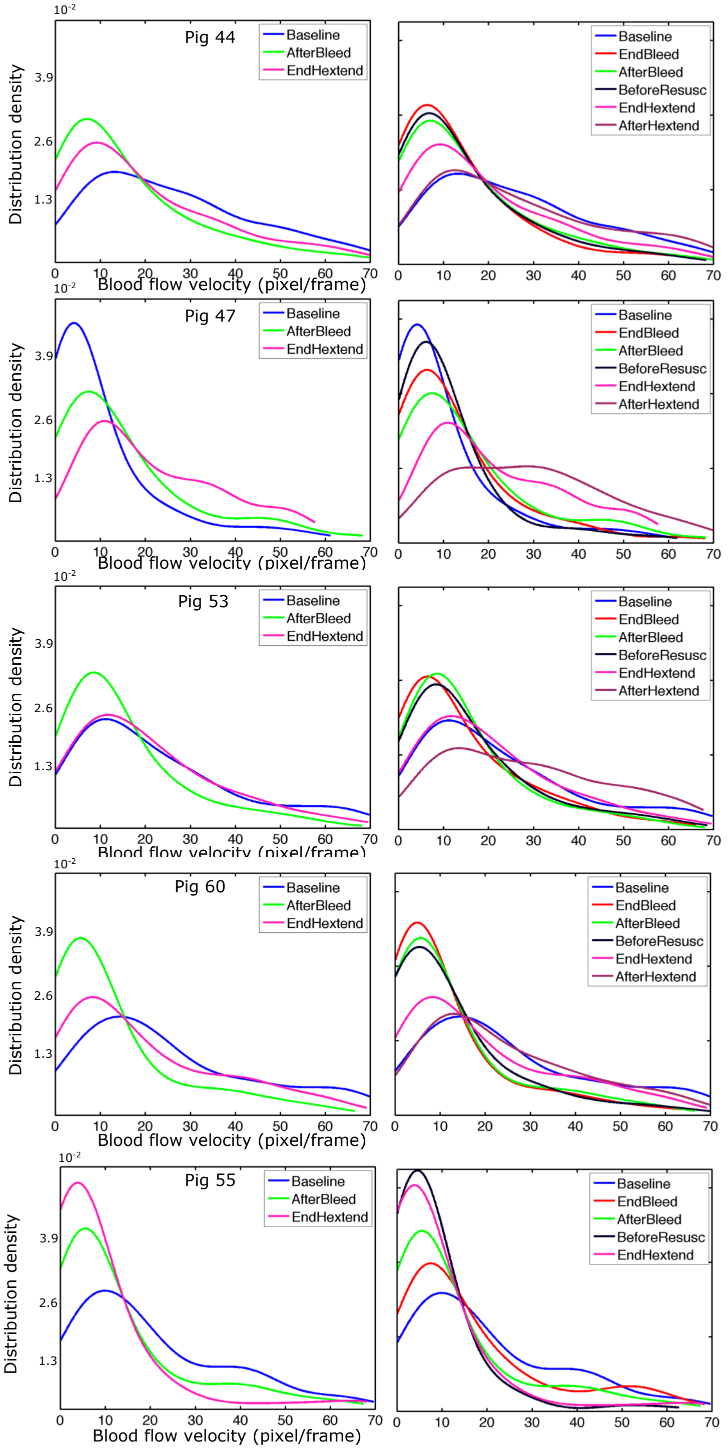
6.6.1 Motion Features

The optical flow based flow estimation fails due to the high-level noise in the captured video and lack of texture around the capillaries. We propose to use the pre-defined spatial-temporal filters to extract the local motions. To detect the spatial-temporal structure of the blood flow, we used the second order derivative of the Gaussian function $G_\theta(x, y, t) = \frac{\partial^2 G}{\partial \theta^2}$ where $G(x, y, t) = e^{-\left(x^2/\sigma_x^2 + y^2/\sigma_y^2 + t^2/\sigma_t^2\right)}$, θ is the direction of the gradient in the spatial-temporal space. In addition to $G_\theta(x, y, t)$, the Hilbert Transforms of the second-order derivatives $H_\theta(x, y, t)$ are included in the filter bank. In our experiment, the filter bank spans 16 spatial orientations corresponding to vessel segments of different directions and 11 temporal orientations corresponding to different blood flow velocity levels.

The spatial-temporal filters suppress the noise given that the noise in the video is random and uncorrelated among different pixels. On the other hand, high response happens if the local flow motion is aligned with the motion pattern of the applied filter. Another benefit of using the spatial-temporal filters is that by adjusting their sizes, we are able to get localized filter responses both spatially and temporally. The prior knowledge about the vessel structures leads us to design the filters of appropriate elongated anisotropic shapes.

With the location denoted as $\mathbf{x}$, the velocity level s and time t, given the filter response $\mathbf{m}(\mathbf{x}, s, t)$, the weighted kernel density of the velocity across all the frames is calculated. For simplicity of denotation, the dependence on the location $\mathbf{x}$ is ignored in the followings. In each frame, the weighted average velocity level $\bar{s}(t)$ is first estimated by:

$$\bar{s}(t) = \frac{\sum_i m(s_i, t)\, s_i}{\sum_i m(s_i, t)} \tag{6.3}$$

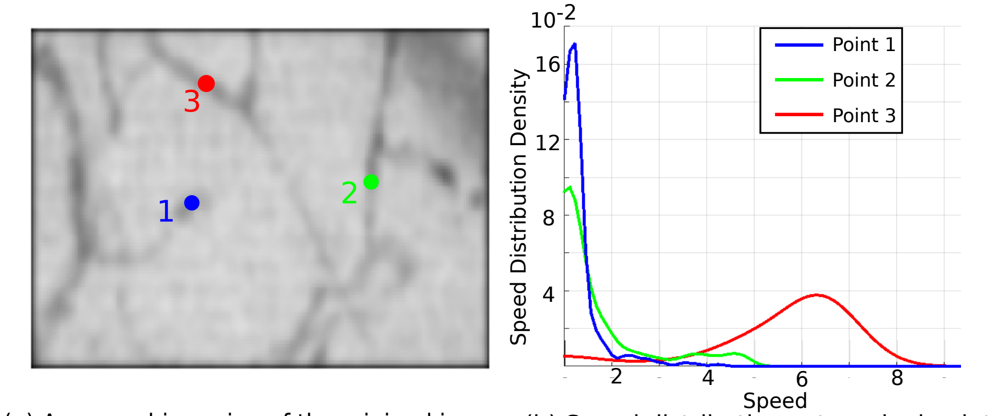


(a) Blood flow velocity distribution at three stages

(b) Blood flow velocity distribution at all six stages

ht

Figure 6.8: The estimated blood flow velocity distributions for pigs at different stages. For each plot, the x-axis is the blood flow velocity, in unit of pixels per frame; the y-axis is the distribution density of vessels with corresponding flow velocity. (a) The blood flow velocity distributions at three key stages of pigs: Baseline, end of bleed and right after resuscitation. (b) The blood flow velocity distributions for all six stages. The annotations for stages: Baseline (blue) - right before the bleeding procedure; EndBleed (red) - end of bleed; Afterbleed (green) - 60 minutes after EndBleed; BeforeResusc (black) - Before resuscitation, 90 minutes after EndBleed, before the resuscitation procedure; EndHextend (purple) - end of resuscitation procedure; AfterHextend (yellow) - 120 minutes after EndHextend.



(a) A zoomed-in region of the minimal image (b) Speed distributions at marked points

Figure 6.9: The velocity level distributions at three points. The corresponded points and velocity level distributions are plotted in the same color. Point *3* is located on a capillary with normal flow. Point *1* and Point *2* are located on capillaries with stopped and intermittent flows respectively.

with $s_i = \{0, 1, \dots 10\}$ for 11 blood flow velocity levels. Based on the average velocity level per-frame The weighted kernel density of the velocity across frames $\hat{f}(s)$ is:

$$f(s) = \frac{\sum_t m(\hat{s}, t)\varphi_h\left(s - \bar{s}(t)\right)}{\sum_t m(\hat{s}, t)} \tag{6.4}$$

where the weight is determined as the motion energy $m\left(\hat{s}, t\right)$, with $\hat{s} = \{0, 1, \dots 10\}$ the closest velocity level to $\bar{s}$; $\varphi_h(x)$ is the kernel function with bandwidth h. In our case we use the Gaussian kernel functions. The weighted kernel density of velocity $f(s)$ is used as the per-pixel motion feature. The motion feature at three locations where different flow types passing by is shown in Fig. (6.9). Point *3* is located on a capillary with normal flow while point *1* and point *2* are located on capillaries with abnormal intermittent flows due to loss of the blood pressure. The intermittency of the flow at point *2* is greater than that for point *1*. This has been reflected in the motion feature in Fig. (6.9) (b): the kernel density for point *2* spans a wider support than point *1*, while almost all density for point *1* is concentrated at a narrow range of velocity level.

6.6.2 The Spatial-temporal filters

The spatial-temporal filters we used in the paper are second derivative of the 3D Gaussian and their Hilbert Transform functions. For the isotropic case, the second derivative with respect to x of the Gaussian is:

$$G_2(x, y, z) = (2x^2 - 1)e^{-\left(x^2+y^2+z^2\right)} \tag{6.5}$$

and its Hilbert Transform is:

$$H_2(x, y, z) = (-2.254x + x^3)e^{-\left(x^2+y^2+z^2\right)} \tag{6.6}$$

To accommodate to the curvilinear structure of the shape of the vessels, we use different scales in the three axis. To describe the flow in different directions and speeds, we use a set of rotated

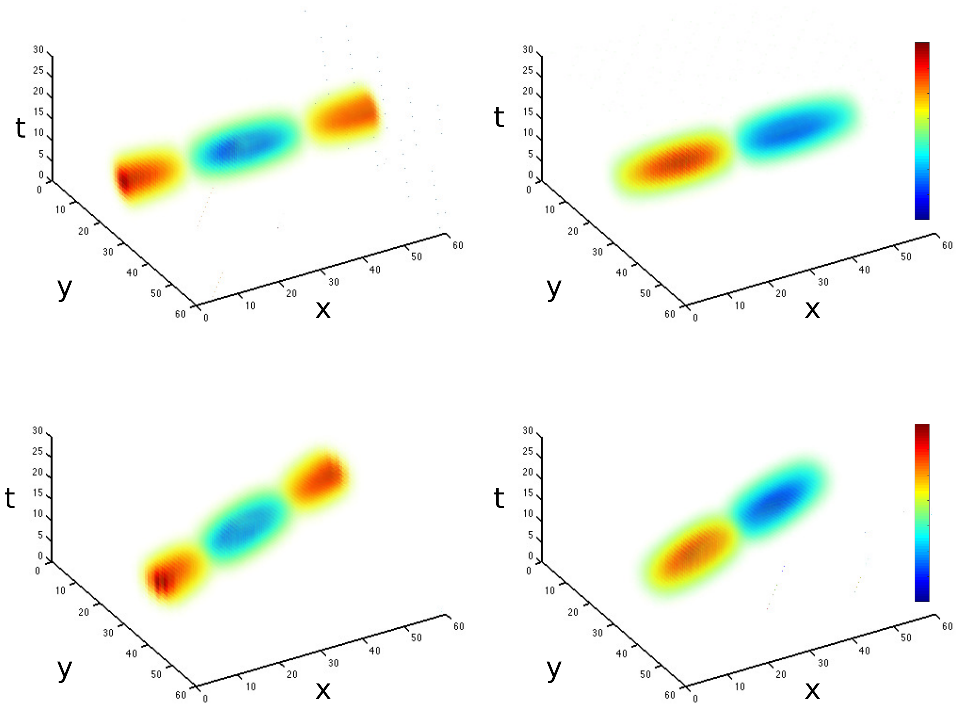


Figure 6.10: Four out of 176 spatial-temporal filters we used in the experiment. Left column: G_2 filters; Right column: H_2 column; First row : the elongated direction of the filters are aligned with the x-axis; Second row : the filters rotated in the x-y plane.

versions of the above defined filters. The filter bank include the filters of 16 spatial orientations and 11 speeds. The visualizations for some of the spatial-temporal filters and the filter responses for a given video are shown in Fig.6.10 and Fig. 6.11.

6.6.3 Blood Flow Types

The blood flow dynamics decreases as a result of the blood pressure reduction during the hemorrhage process. To better quantify and visualize the flow motion pattern changes, based on the clinical experience we define three types of flows: stopped flow, intermittent flow and normal flow. For the stopped flow, the blood within the capillaries has little or no motion either because the blood pressure is insufficient or due to the external pressure introduced by the contact with the measurement device. The intermittent flow includes the flows with unstable velocity. Usually it varies within the low velocity range. The normal flow shows fast and consistent motion patterns. According to the definitions, the velocity distributions $f(s)$ for normal and stopped flows are concentrated within a high velocity range and the range close to zero; while for the intermittent flow, the velocity distribution is similar to the stopped flow but spans a wider velocity range due to the intermittency in the flow motion, as shown in Fig. 6.9.

6.6.4 Classify the Blood Flow Types

Although we are able to quantify the match of the current flow to either of the three defined types based on the scoring functions defined above, it is not guaranteed that the three flow types are mutually exclusive since the scoring functions are defined separately and in a heuristic manner. So in addition to using the manually defined scoring functions, we also propose a learning-based approach with cascade classifiers. To this end, we have labeled all the 97 micro-circulatory videos of the 18 pigs in the experiment. In each video, a subset of vessels/background area are labeled as one of *stopped*, *intermittent*, *normal* flows and *background* classes. Examples of the labeled video are included in the supplementary material.

In the first stage, vessels are separated from the background based on the local structure information encoded in the spatial structural feature:

$$\mathbf{l}_1 = \left[I_{\text{min}}, \sigma_t, f_{\text{OOF}}\left(k_i; I_{\text{min}}\right), \ldots, f_{\text{OOF}}\left(k_i; \sigma_t\right) \ldots\right] \tag{6.7}$$

, where I_{min} is the minimal image, σ_t is the intensity variance map across all frames $f_{\text{OOF}}(k_i; I_{\text{min}})$ is the Optimal Oriented Flux filter [103] response at scale k_i operated on I_{min}.

In the second stage, wide vessels are removed and the flow patterns in those wide vessels are not considered as capillaries. The analysis of blood flow is therefore restricted to capillaries less than 20 μm in diameter, because these are the vessels involved in oxygen exchange and thus in tissue perfusion. The larger vessels can be used to evaluate for possible measurement artifacts such as excessive pressure. Also, since the wider vessels are located deeper below the surface than the capillaries, they are usually out-of-focus given the small depth of focus of a micro-scale lens. Thus the evaluated motion patterns in the wide vessels are not reliable due to the blurring effect. In the second stage, we use the same feature as in the first stage to represent the local structure information.

In the third stage, the blood flow within the detected vessels are categorized into stopped, intermittent and normal flows. The features for flow type determination is the concatenation of speed level distribution evaluated using Eq. 6.4 along with the local structural features:

$$\mathbf{l}_3 = \left[\mathbf{f}(s), f_{\text{OOF}}\left(k_i; I_{\text{min}}\right), \ldots, f_{\text{OOF}}\left(k_i; \sigma_t\right) \ldots\right] \tag{6.8}$$

,where $\mathbf{f}$ is the speed distribution defined in Eq. 6.4. We use the Random Forest Classifiers in all three stages.

Compared with a one-stage classifier which directly categorize the pixels into background and three types of flows, the cascade classifier emphasizes different types of features in stages. For the task of separating the vessels from the background, the statistics of the video such as the denoised vessel enhanced image provides more structural information than the raw frames from the video with high-level noise and lack of texture.

To evaluate the robustness of the features and the learned classifiers, we train and test the cascade classifiers in three cases with different rules of selecting the training and testing set: (1) The training and testing samples are selected randomly from the labeled data without any constraints; (2) The samples are selected such that the training and testing samples are on different vessels; (3) Training and testing samples are selected from videos of different pigs. In the second and third cases, we consider the influence of the variance in locations and subjects. The performance is evaluated in terms of the third stage in the cascade classifier and shown in Fig. 6.12. For all classes, although the performance drops due to the variances among vessels and different subjects as expected, those drops are relatively small. This suggests some level of robustness of the proposed approach to inter-subject variability.

The blood flow type map estimated by the cascade classifier for Pig 54 is shown in Fig. 6.13. The local changes in the types of flow due to bleeding and resuscitation are shown to be different for different locations. This local flow type measurement provides a new approach for the clinicians to study the oxygen delivery status in micro-scale.

6.7 Conclusion

We presented a multi-stage framework for processing microcirculatory videos automatically and in real time. The processing stages include video stabilization, image enhancement, and micro-vessel extraction, in order to automatically estimate statistics of the micro blood flow captured in SDF videos. We applied our method to analyze changes in microcirculation in test animals at different stages of induced bleeding experiment, including before, during and after bleeding as well as after resuscitation. The results show that by using image augmentation and continuous video sampling techniques, reliable microcirculatory imaging processing can be automated and accomplished in real time despite the inherent challenges to microcirculatory flow quantization. The parameters described in this analysis represent novel metrics of SDF imaging that should substantially improve the utility of SDF imaging to assess microcirculatory changes with disease and its treatment. In addition, local features such as local flow velocity variation and intermittency of the flow have been studied to further enhance the functionality and clinical relevance of the framework.

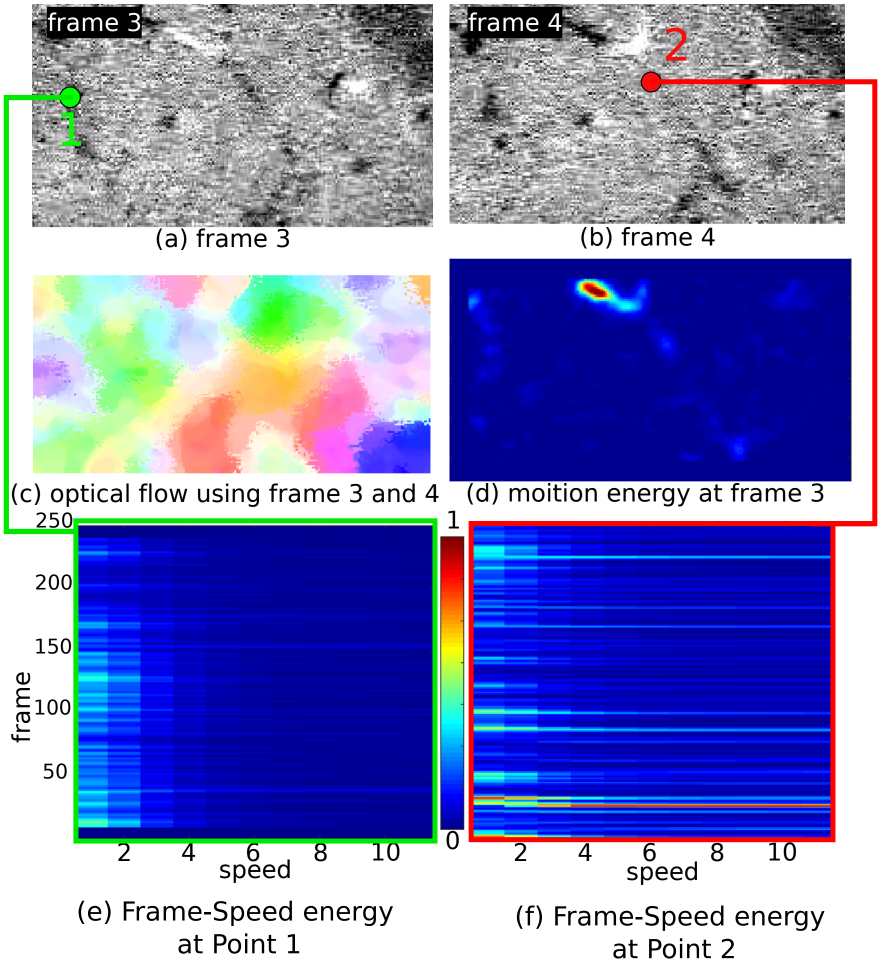
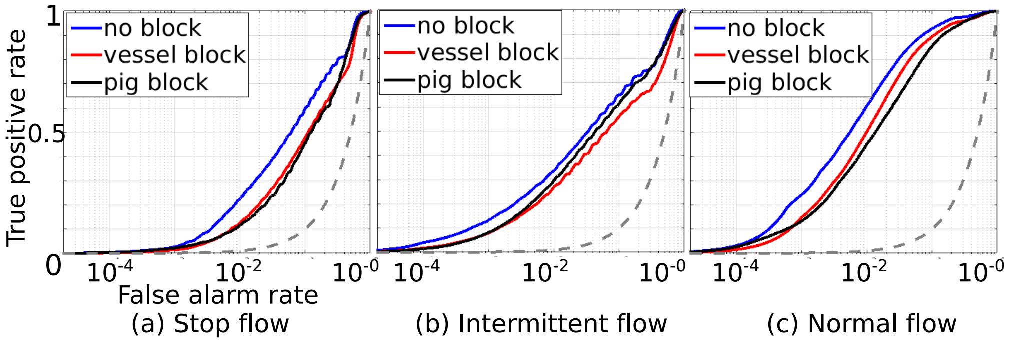

frame 3
1
frame 4
2
(a) frame 3
(b) frame 4
(c) optical flow using frame 3 and 4
(d) moition energy at frame 3
250
200
150
100
50
frame
2
4
6
8
10
speed
1
0
2
4
6
8
10
speed
(e) Frame-Speed energy at Point 1
(f) Frame-Speed energy at Point 2

Figure 6.11: Local motion estimations on a zoomed-in region of the motion magnified video showing the micro-vascular blood flow. (a) The zoomed-in region in frame 3 of the motion magnified video; (b) The same local region as in (a) in frame 4 ; (c) The estimated optical flow using the two frames in (a) and (b); (d) The overall motion energy calculated from the filter response. The flow motion induces high motion energy while the motion energy for the noise is low since the artifact motion patterns introduced by noise are not aligned with any of the applied 3D filters. (e)(f) The filter responses at the marked points in (a), revealing the approximate speed of the flow and the key frames in which there is observable blood flow passing by those locations. The flow speed at Point 2 is shown to be faster than Point 1.



(a) Stop flow (b) Intermittent flow (c) Normal flow

Figure 6.12: The ROCs evaluating the flow type classification on the testing sets for three cases of train/test sample selection: No block - The training and testing samples are selected randomly from the labeled data without any constraints; Vessel block - the samples are selected such that the training and testing samples are from different vessels; Pig block - the samples are selected such that the training and testing samples are from videos of different subjects. (a) The performance on the stopped flow; (b) The performance on the intermittent flow; (c) The performance on the normal flow.

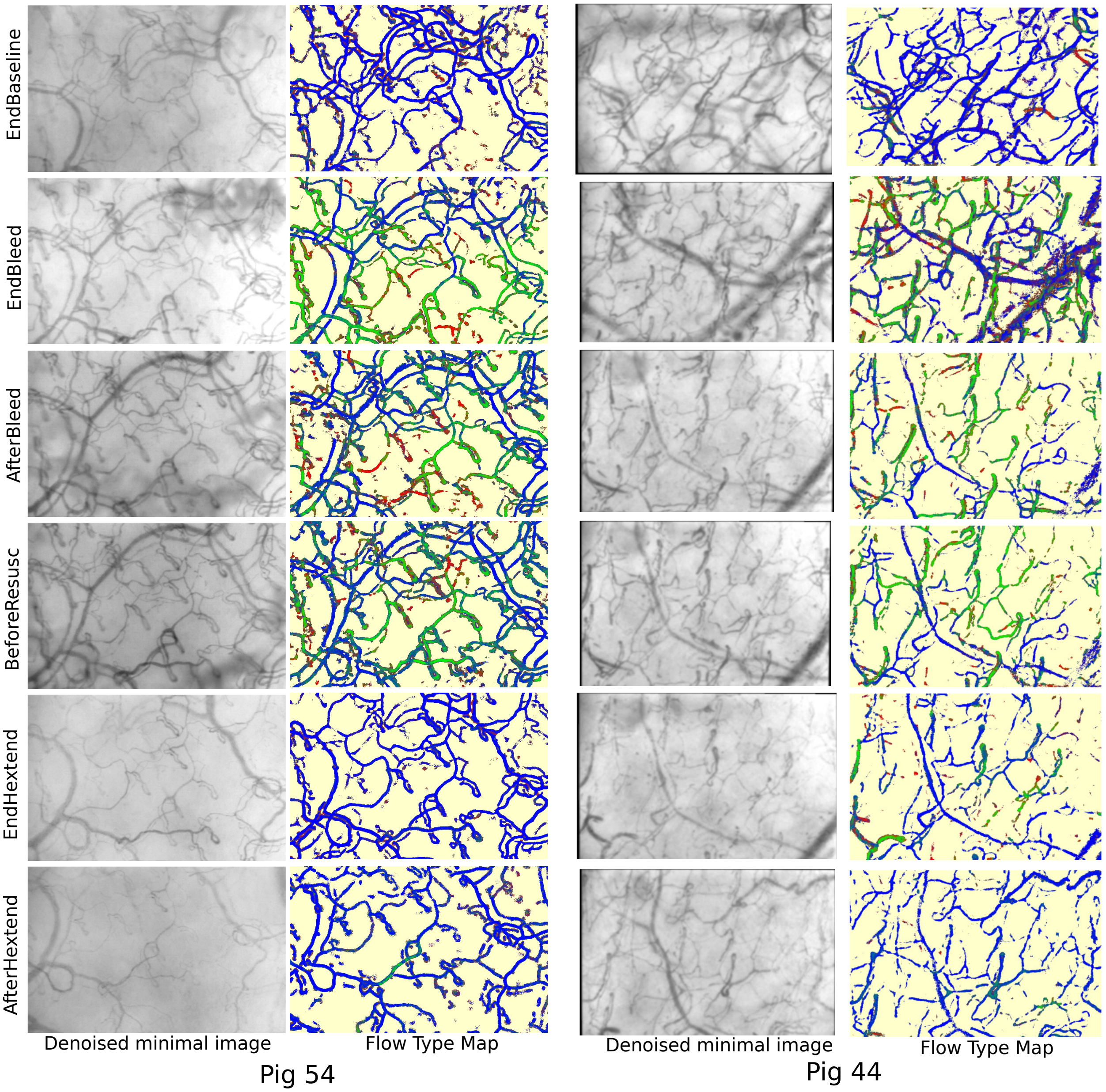


Figure 6.13: The blood flow type map estimated by the cascade classifier for Pig 54 at different different critical care stages. The stages are defined in Sec. 6.5. The color encoding for flow types: red - stopped flow; green - intermittent flow; blue - normal flow; light yellow - background. The dynamics of the blood flow in the capillaries decreases, manifesting as increased fraction of stopped and intermittent flows, during the hemorrhage process. Then it recovers to normal after the resuscitation process.

Chapter 7

Conclusion

In this thesis, we work towards the goal of developing computational methods and small baseline imaging system for 3D sensing of complex scenes in real world conditions, with the design principle of physically modeling the scene complexities and specifically inferring the uncertainties for the images captured with small baseline setups. We have shown that the challenges in the real world condition can be tackled by physically modeling the way light interacts with the scene and specifically inferring the uncertainty.

In Chapter 2, we introduce a compact photometric stereo system using a small LED ring (6 cm in diameter) as the light sources. By utilizing the differential images, we show that the highly non-convexity of the original inverse problem can be greatly alleviated. With the proposed method, we are able to reconstruct high quality 3D mesh with a compact and low-cost imaging system.

In Chapter 3, a matting and depth estimation method using a focal stack image has been discussed for reconstructing scenes with high spatial frequency and mutual occlusions. The method has been applied for in-vivo micron scale reconstruction of capillary veins.

In Chapter 4, we develop a learning based pipeline for monocular depth estimation from a monocular video with uncertainties. The per-frame depth probability distribution is fused over frames in a Bayesian way. This not only leads to an accurate and temporally consistent depth sensing scheme, but also an uncertainty estimation that can be useful for various applications.

In Chapter 5, we implement a high resolution DOT imaging pipeline with a pair of high resolution camera and laser projector. The verged projector-camera setup enables the capturing of short-range indirect images over smaller FoVs. Hence more details about the fine structure underneath the scatter medium are imaged. The scanning lines setup, rather than the paired points setup in traditional DOT, enables a new highly efficient 3D tomography algorithm.

In Chapter 6, we showcase a fully automated real-time system for analyzing the blood flow within capillary veins from a microscopic video. The video is captured with a Sidestream Dark Field imaging device consisting of a video camera and a small ring of LEDs. The analysis system can serve as a tool to greatly reduce the manual blood flow analysis efforts for critical care doctors.

7.1 Future Work

There are several directions in which we can extend the computational methods and hardware developed throughout this thesis. Those directions include more physically accurate models of the scene complexity, optimized representation for the 3D structure, theoretical modeling for estimation uncertainties, and more agile subsurface imaging. These directions lead to the following specific topics for future works.

Appearance capture of more general scenes In Chapter 2, we have assumed that the object surface is Lambertian. However, this assumption is not held for a lot of materials in the real world. Although the usage of small baseline light source has partially alliterated the failure of this assumption, there are still regions on the object where the lines of sight are close to the glazing angle of the surface. As a result, the specular lobe will be present in most images captured with a close cluster of light sources. As a result, the image formation model used for our surface reconstruction pipeline will fail to model the light intensity. One way to tackle this issue is to use a polarizer in front of the camera to remove the specular lobes so the remaining component can be approximated using the Lambertian model [172]. An alternative way is to explicitly model the BRDF of the material and recover both the BRDF and the geometry of the surface, or derive the BRDF invariants [26] for near-light photometric stereo and extract the geometry information with the invariant constraint.

For scatter medium, the light reaching the camera is contributed by both the direct reflectance from the surface, and the scattered light from points other than the sensed point. In general, the subsurface scattering effect acts as a low pass filter that blurs the image gradient. As a result, the reconstructed 3D surface without considering the subsurface scatter is smoother than the true shape, with sharp curvatures blurred out. One way to alleviate this issue is to model the subsurface scattering effect with a convolution over the surface and perform the deconvolution after an initial guess of the 3D geometry is available. One thing to note here is that the deconvolution operation has to be applied on the 2D manifold of the surface rather than directly in the image domain.

Last but not least, for real-world application, it is necessary to consider multiple rather than one single object in the scene. One example of application is the 3D reconstruction of a room (or part of it) using the NIR LED light sources around an indoor surveillance camera. In this case, the depth variations among objects would be larger than those for the objects itself. In addition, the cast-shadows among different objects need to be considered. The indoor scene 3D shape recovery using small baseline light sources will be useful for surveillance-related tasks such as background subtraction and abnormal detection.

3D sensing with semi-transparent or volume occlusions In Chapter 3, we have modeled the occlusion as completely opaque. For occlusions such as rain drops, stained glass, and tissues commonly seen in daily photography and microscopy, the foreground occlusions are semi-transparent, allowing part of the background light go through the occluder and reach the camera. Simultaneous matting and depth estimation for semi-transparent occluders will be a more challenging task than its opaque counterpart, since the degree of transparency serves as another set of spatially-variant variables, in addition to the matting pattern and depth values. To handle the additional set of variables for the semi-transparent case while maintaining the complexity of the inverse problem, we can consider using single imaging matting methods [210] to decompose the problem into two separate problems: matting pattern estimation, and depth estimation for the foreground occlusions.

Other than representing the occlusions as layer-wise depth maps, a more realistic geometric model for the occlusions should include the thickness. For example, the shape of the projection of a 3D sphere onto an image plane would be different from the shape of a 2D circle projection. Octree-based volumetric representation can be used for memory efficiency and 3D structure with higher resolution.

Learning 3D representation tailored for small baseline setups Compared with predefined 3D representation such as layer-wise depth voxel grid, (truncated) signed distance function, the CNN-encoded volumetric representation [144, 171] has shown to be both compact and effective in representing high-resolution 3D structure. The encoders are learned with images captured with sparely sampled view points. For small camera baseline images (*i.e.*densely angular sampling for camera views), more information about the reflection property of the surface is available, especially for non-Lambertian materials. As a result, it would be an promising direction to learn the encoder for both 3D structure and the surface at the same time, by using images/videos from densely angular sample views.

Another direction is to learn the 3D encoder for a scene rather than a single 3D object. To regularize the huge configuration space of the 3D scene, besides using a simplified raw representation for 3D (*e.g.*depth map as used in [15]), we can utilize the graph representation to encode the relation for multiple objects in the scene along with the single object 3D encoder to enjoy the benefits from both world. Compared with 3D scene encoding for a single object, learning the optimal representation for a 3D scene has more applications, such as view interpolation/extrapolation and scene relighting.

Extending and utilizing the uncertainties In Chapter 4, we have shown that the uncertainty map can be used for filling the regions with inaccurate depth and low estimation confidence to generate more accurate depth map. In addition to serving as a proxy for post-processing, the uncertainty estimation can be used for other purposes. For example, by classifying the source of uncertainty (*e.g.*due to lack of input information, or due to lack of the representation ability of the model), we can tell the direction to improve the model in order to have more reliable depth sensing results. As another example, for regions with lower confidence, we can apply the single image depth estimation method, which is usually more computationally expensive, but more robust to the cases where the triangulation based methods fail. Also, rather than focusing on the depth uncertainty estimation, it would be useful to model the probabilistic distribution of other unknowns such as camera pose or the 3D scene flow for dynamic scenes.

Another interesting direction is to guide the sampling distribution for a dedicated depth sensor by taking advantage of the uncertainty from a helper RGB camera [146]. Depth sensors such as LiDAR usually have very long sensing range and high depth resolution. But due to the complex sensor fabrication procedure, the spatial resolution is much lower than that for an RGB sensor. By re-distributing the depth sampling points of the depth sensor over regions with lower depth confidence for an RGB camera, we can have a hybrid imaging system with both high accuracy in depth measurement, and high spatial resolution from the RGB camera at the same time.

Faster EpiVerge system with more degrees of freedom Currently the capture speed of Epiverge is limited due to the usage of the point-to-point (quasi static) mode of the MEMS for projecting vector graphs. To speed up the capture procedure for dynamic scene imaging, we need to replace the point scanning MEMS laser with a line scanning laser system. The line scanning can be implemented with a Powell lens that reshapes a laser spot into a plane. The reshaped plane illumination is further reflected and redirected with a 1D galvo mirror. In this way, we can generate a pencil of laser lines by rotating the galvo. The frame rate for the laser project is determined by the 1D galvo mirror. In this case, the epipolar geometry is formulated by the camera image plane, and the position and orientation of the 1D galvo mirror. Their relative pose could be calibrated with the same calibration process for the current system.

More control dimensions, such as spatial coding and light source spectrum, could be added on top of the current system. To add spatial coding, we can simply place another LCoS or DMD to add spatial patterns. To add control over the light source spectrum, we can use a programmable spectral light source rather than a narrow band laser with fixed wavelength [156].

Bibliography

[1] https://www.luminartech.com/technology. 1.2.2

[2] https://www.ese.wustl.edu/~nehorai/research/dot/dot_overview.html. 5.1

[3] Robust Vision Challenge Workshop. http://www.robustvision.net, 2018. 4.1, 4.2, 4.4

[4] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE Trans. Pattern Anal. Mach. Intell.*, 34(11):2274–2282, Nov. 2012. 3.5

[5] S. Agarwal, K. Mierle, and Others. Ceres solver. http://ceres-solver.org. 2.6.1

[6] S. R. Arridge and J. C. Schotland. Optical tomography: forward and inverse problems. *Inverse Problems*, 25(12):123010, dec 2009. 5.2

[7] J. T. Barron and J. Malik. Shape, illumination, and reflectance from shading. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(8):1670–1687, 2015. 2.6.3

[8] J. T. Barron and B. Poole. The fast bilateral solver. In *European Conference on Computer Vision (ECCV)*, 2016. 4.13

[9] R. Basri, D. Jacobs, and I. Kemelmacher. Photometric stereo with general, unknown lighting. *International Journal of Computer Vision*, 72(3):239–257, 2007. 1.2.2

[10] S. Belanger, M. Abran, X. Intes, C. Casanova, and F. Lesage. Real-time diffuse optical tomography based on structured illumination. *Journal of Biomedical Optics*, 15(1):016006, 2010. 5.2

[11] P. N. Belhumeur and D. J. Kriegman. What Is the Set of Images of an Object Under All Possible Illumination Conditions ? *International Journal*, 260(28):245–260, 1998. 2.2

[12] P. N. Belhumeur, D. J. Kriegman, and A. L. Yuille. The Bas-Relief Ambiguity. *International Journal of Computer Vision*, 35(1):33–44, 1999. 2.2

[13] B. Bi, B. Han, W. Han, J. Tang, and L. Li. Image reconstruction for diffuse optical tomography based on radiative transfer equation. *Computational and Mathematical Methods in Medicine*, 2015:1–23, 2015. 5.2

[14] C. M. Bishop. Mixture density networks. 1994. 4.3

[15] M. Bloesch, J. Czarnowski, R. Clark, S. Leutenegger, and A. Davison. CodeSLAM - Learning a compact, optimisable representation for dense visual SLAM. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 4.1, 4.2, 7.1

[16] D. A. Boas. *Diffuse photon probes of structural and dynamical properties of turbid media: theory and biomedical applications*. PhD thesis, University of Pennsylvania, 1996. 5.2

[17] D. A. Boas, J. P. Culver, J. J. Stott, and A. K. Dunn. Three dimensional monte carlo code for photon migration through complex heterogeneous media including the adult human head. *Optics Express*, 10(3):159, feb 2002. 5.2, 5.3, 5.3, 5.6.1

[18] S. Brahmbhatt, J. Gu, K. Kim, J. Hays, and J. Kautz. Geometry-aware learning of maps for camera localization. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 1.2.1

[19] J. D. Briers and S. Webster. Laser speckle contrast analysis (LASCA): a nonscanning, full-field technique for monitoring capillary blood flow. . *J. Biomed. Opt*, Apr. 1996. 6.2

[20] S. Blanger, M. Abran, X. Intes, C. Casanova, and F. Lesage. Real-time diffuse optical tomography based on structured illumination. *Journal of Biomedical Optics*, 15(1):1 – 7, 2010. 5.1

[21] K. M. Case, P. F. Zweifel, and G. C. Pomraning. Linear transport theory. *Physics Today*, 21(10):72–73, oct 1968. 5.3

[22] M. Chalia, L. A. Dempsey, R. J. Cooper, C.-W. Lee, A. P. Gibson, J. C. Hebden, and T. Austin. Diffuse optical tomography for the detection of perinatal stroke at the cot side: a pilot study. *Pediatric Research*, 85(7):1001–1007, feb 2019. 5.1

[23] D. Chan, H. Buisman, C. Theobalt, and S. Thrun. A noise-aware filter for real-time depth upsampling. In *Workshop on Multi-camera and Multi-modal Sensor Fusion Algorithms and Applications - M2SFA2 2008*, Marseille, France, 2008. Andrea Cavallaro and Hamid Aghajan. 4.1, 4.2

[24] M. Chandraker. The information available to a moving observer on shape with unknown, isotropic brdfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(7):1283–1297, July 2016. 1

[25] M. Chandraker, J. Bai, and R. Ramamoorthi. On differential photometric reconstruction for unknown, isotropic brdfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(12):2941–2955, Dec 2013. 1, 1.3

[26] M. Chandraker, J. Bai, and R. Ramamoorthi. On differential photometric reconstruction for unknown, isotropic BRDFs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(12):2941–2955, 2013. 2.2, 2.4, 7.1

[27] S. Chandrasekhar. *Radiative Transfer*. International series of monographs on physics. Dover Publications, 1960. 1.3

[28] J.-R. Chang and Y.-S. Chen. Pyramid stereo matching network. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5410–5418, 2018. 1.2.1, 4.1, 4.2, 4.3.1, 4.6

[29] Chao Liu, H. Gomez, S. Narasimhan, A. Dubrawski, M. R. Pinsky, and B. Zuckerbraun. Real-time visual analysis of microvascular blood flow for critical care. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2217–2225, 2015. 1.4

[30] C. Chen, F. Tian, H. Liu, and J. Huang. Diffuse optical tomography enhanced by clustered sparsity for functional brain imaging. *IEEE Transactions on Medical Imaging*, 33(12):2323–2331, dec 2014. 5.2

[31] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *arXiv:1606.00915*, 2016. 1.3

[32] H. Cheng, Q. Luo, Q. Liu, Q. Lu, H. Gong, and S. Zeng. Laser speckle imaging of blood flow in microcirculation. . *Phys. Med. Bio*, 2004. 6.2

[33] D. Cho, Y. Matsushita, Y.-W. Tai, and I.-S. Kweon. Photometric Stereo Under Non-uniform Light Intensities and Exposures. *ECCV*, pages 170–186, 2016. 2.2

[34] J. A. Christian and S. Cryan. A survey of LiDAR technology and its use in spacecraft relative navigation. In *AIAA Guidance, Navigation, and Control (GNC) Conference*, 2013. 4.1, 4.2

[35] J. Clark. Active Photometric Stereo. In *CVPR*, 1992. 1.3, 2.2

[36] R. Clark, S. Wang, A. Markham, N. Trigoni, and H. Wen. VidLoc: a deep spatial-temporal model for 6-DoF video-clip relocalization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 4.2

[37] S. B. Colak, D. G. Papaioannou, G. W. 't Hooft, M. B. van der Mark, H. Schomberg, J. C. J. Paasschens, J. B. M. Melissen, and N. A. A. J. van Asten. Tomographic image reconstruction from optical projections in light-diffusing media. *Applied Optics*, 36(1):180, jan 1997. 5.4

[38] A. Corlu, R. Choe, T. Durduran, M. A. Rosen, M. Schweiger, S. R. Arridge, M. D. Schnall, and A. G. Yodh. Three-dimensional in vivo fluorescence diffuse optical tomography of breast cancer in humans. *Opt. Express*, 15(11):6696–6716, May 2007. 1.3

[39] A. Criminisi, I. Reid, and A. Zisserman. Single view metrology. *International Journal of Computer Vision (IJCV)*, 2000. 4.2

[40] B. Cubelos, A. Sebastin-Serrano, L. Beccari, M. E. Calcagnotto, E. Cisneros, S. Kim, A. Dopazo, M. Alvarez-Dolado, J. M. Redondo, P. Bovolenta, C. A. Walsh, and M. Nieto. Cux1 and cux2 regulate dendritic branching, spine morphology, and synapses of the upper layer neurons of the cortex. *Neuron*, 66(4):523 – 535, 2010. 3.1

[41] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner. ScanNet: Richly-annotated 3D reconstructions of indoor scenes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1.2.1, 4.10, 4.4, 4.4

[42] D. De Backer, S. Hollenberg, C. Boerma, P. Goedhart, G. Büchele, G. Ospina-Tascon, I. Dobbe, and C. Ince. How to evaluate the microcirculation: report of a round table conference. *Critical Care*, 11(5):R101, 2007. 6.5

[43] J. G. Dobbe, G. J. Streekstra, B. Atasever, R. van Zijderveld, and C. Ince. Measurement of functional microcirculatory geometry and velocity distributions using automated image analysis. *Medical & biological engineering & computing*, 46(7):659–670, Apr. 2008. 1.1, 6.2

[44] B. Dong, K. D. Moore, W. Zhang, and P. Peers. Scattering parameters and surface normals from homogeneous translucent materials using photometric stereo. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2299–2306, 2014. 2.2

[45] C. S. Donner. *Towards Realistic Image Synthesis of Scattering Materials*. PhD thesis, La Jolla, CA, USA, 2006. AAI3226771. 1.3, 1

[46] J. Dorsey, A. Edelman, H. W. Jensen, J. Legakis, and H. K. Pedersen. Modeling and rendering of weathered stone. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '99, pages 225–234, New York, NY, USA, 1999. ACM Press/Addison-Wesley Publishing Co. 1

[47] D. Droeschel, D. Holz, and S. Behnke. Multi-frequency phase unwrapping for time-of-flight cameras. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1463–1469, 2010. 1.2.2

[48] T. Durduran, R. Choe, W. B. Baker, and A. G. Yodh. Diffuse optics for tissue monitoring and tomography. *Reports on Progress in Physics*, 73(7):076701, jun 2010. 5.2, 5.3

[49] S. K. S. G. E. Alexander, Q. Guo and T. Zickler. Depth from focus with your mobile phone. In *ECCV*, October 2016. 3.1

[50] A. Edmans and X. Intes. Mesh optimization for monte carlo-based optical tomography. *Photonics*, 2(2):375–391, apr 2015. 5.2

[51] D. Eigen, D. Krishnan, and R. Fergus. Restoring an image taken through a window covered with dirt or rain. In *ICCV*, pages 633–640, 2013. 1.3, 3.2

[52] D. Eigen, C. Puhrsch, and R. Fergus. Depth map prediction from a single image using a multi-scale deep network. In *Advances in Neural Information Processing Systems (NIPS)*, 2014. 4.4, 4.1, 4.4, 4.2, 4.4

[53] J. Engel, V. Koltun, and D. Cremers. Direct sparse odometry. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 40:611–625, 2018. 1.2.1, 4.3.4, 4.4

[54] J. B. Evers, A. R. van der Krol, J. Vos, and P. C. Struik. Understanding shoot branching by modelling form and function. *Trends in Plant Science*, 16(9):464 – 467, 2011. 3.1

[55] P. Favaro, S. Louis, and L. Angeles. Seeing Beyond Occlusions (and other marvels of a finite lens

aperture). In *CVPR*, pages 1–8, 2003. 3.2

[56] J. Fiss, B. Curless, and R. Szeliski. Light Field Layer Matting. In *CVPR*, 2015. 3.1, 3.2

[57] A. Fix, A. Gruber, E. Boros, and R. Zabih. A graph cut algorithm for higher-order markov random fields. In *ICCV*, pages 1020–1027, Nov 2011. 3.4

[58] A. F. Frangi, W. J. Niessen, K. L. Vincken, and M. A. Viergever. Multiscale vessel enhancement filtering. *MICCAI*, pages 130–137, 1998. 6.3.3

[59] J. R. Frisvad, T. Hachisuka, and T. K. Kjeldsen. Directional dipole model for subsurface scattering. *ACM Trans. Graph.*, 34(1):5:1–5:12, Dec. 2014. 1.3, 1

[60] H. Fu, M. Gong, C. Wang, K. Batmanghelich, and D. Tao. Deep ordinal regression network for monocular depth estimation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 1.2.1, 1.3, 4.1, 4.1, 4.2, 4.4, 4.1, 4.2, 4.3, 4.4, 4.11, 4.12

[61] H. Fujii, Y. Yamada, K. Kobayashi, M. Watanabe, and Y. Hoshi. Modeling of light propagation in the human neck for diagnoses of thyroid cancers by diffuse optical tomography. *International Journal for Numerical Methods in Biomedical Engineering*, 33(5):e2826, oct 2016. 5.1

[62] A. Gaidon, Q. Wang, Y. Cabon, and E. Vig. Virtual worlds as proxy for multi-object tracking analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 1.2.1, 4.4, 4.4

[63] Y. Gal and Z. Ghahramani. Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In *International Conference on Machine Learning (ICML)*, 2016. 4.1, 4.2

[64] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? The KITTI vision benchmark suite. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3354–3361, 2012. 1.2.1, 4.4, 4.2, 4.4

[65] A. Ghosh, G. Fyffe, B. Tunwattanapong, J. Busch, X. Yu, and P. Debevec. Multiview face capture using polarized spherical gradient illumination. *Proceedings of the 2011 SIGGRAPH Asia Conference on - SA '11*, 30(6):1, 2011. 2.2

[66] S. Gioux, A. Mazhar, and D. J. Cuccia. Spatial frequency domain imaging in 2019: principles, applications, and perspectives. *Journal of Biomedical Optics*, 24(07):1, jun 2019. 5.2

[67] C. Godard, O. Mac Aodha, and G. J. Brostow. Unsupervised monocular depth estimation with left-right consistency. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1.2.1, 4.1, 4.1, 4.2, 4.4, 4.2, 4.4

[68] A. Godavarty, S. Rodriguez, Y.-J. Jung, and S. Gonzalez. Optical imaging for breast cancer pre-screening. *Breast Cancer: Targets and Therapy*, page 193, jul 2015. 5.1

[69] P. T. Goedhart, K. M, R. Bezemer, J. Merza, and C. Ince. Sidestream Dark Field (SDF) imaging:a novel stroboscopic LED ring-based imaging modality for clinical assessment of the microcirculation. In *Optics Express*, pages 1–14, Dec. 2007. 6.1, 6.1, 6.2

[70] D. B. Goldman, B. Curless, A. Hertzmann, and S. M. Seitz. Shape and spatially-varying brdfs from photometric stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(6):1060–1071, 2010. 1.2.2

[71] H. Gómez, J. Mesquida, L. Hermus, P. Polanco, H. Kim, S. Zenker, A. Torres, R. Namas, Y. Vodovotz, G. Clermont, J. Puyana, and M. Pinsky. Physiologic responses to severe hemorrhagic shock and the genesis of cardiovascular collapse: can irreversibility be anticipated? . *Journal of Surgical Research*, Nov. 2012. 6.1, 6.5

[72] P. Graham, B. Tunwattanapong, J. Busch, X. Yu, A. Jones, P. Debevec, and A. Ghosh. Measurement-based synthesis of facial microgeometry. *Computer Graphics Forum*, 32(2 PART3):335–344, 2013. 2.2

[73] J. Gu, R. Ramamoorthi, P. Belhumeur, and S. Nayar. Removing image artifacts due to dirty camera lenses and thin occluders. *ACM Transactions on Graphics (TOG)*, 2009. 1.3, 3.1, 3.2, 3.3

[74] S. Gupta, R. Girshick, P. Arbelaez, and J. Malik. Learning rich features from RGB-D images for

object detection and segmentation. In *European Conference on Computer Vision (ECCV)*, 2014. 4.1, 4.2

[75] M. Guven, B. Yazici, X. Intes, and B. Chance. Diffuse optical tomography with a priori anatomical information. *Physics in Medicine and Biology*, 50(12):2837–2858, jun 2005. 5.2

[76] A. Handa, V. Ptrucean, S. Stent, and R. Cipolla. Scenenet: An annotated model generator for indoor scene understanding. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5737–5743, 2016. 1.2.1

[77] S. Hasinoff and K. Kutulakos. Multiple-Aperture Photography for High Dynamic Range and Post-Capture Refocusing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2009. 1.2.1, 3.1, 3.2, 3.3

[78] S. W. Hasinoff and K. N. Kutulakos. A Layer-Based Restoration Framework for Variable-Aperture Photography. In *ICCV*, 2007. 1.2.1, 3.1, 3.2, 3.3

[79] M. S. Hassouna and A. A. Farag. MultiStencils Fast Marching Methods: A Highly Accurate Solution to the Eikonal Equation on Cartesian Domains. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(9):1563–1574, 2007. 6.2

[80] K. He, J. Sun, and X. Tang. Single image haze removal using dark channel prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(12):2341–2353, 2011. 1.3

[81] L. C. Henyey and J. L. Greenstein. Diffuse radiation in the galaxy. *The Astrophysical Journal*, 93:70, jan 1941. 5.3

[82] R. Horaud, M. Hansard, G. Evangelidis, and C. Ménier. An overview of depth cameras and range scanners based on time-of-flight technologies. *Machine Vision and Applications Journal*, 27(7):1005–1020, 2016. 4.1, 4.2

[83] J. Hu, M. Ozay, Y. Zhang, and T. Okatani. Revisiting single image depth estimation: Toward higher resolution maps with accurate object boundaries. 03 2018. 1.3

[84] B. Huang, W. Wang, M. Bates, and X. Zhuang. Three-dimensional super-resolution imaging by stochastic optical reconstruction microscopy. *Science*, 319(5864):810–813, 2008. 3.1

[85] P.-H. Huang, K. Matzen, J. Kopf, N. Ahuja, and J.-B. Huang. DeepMVS: Learning multi-view stereopsis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 4.1, 4.2

[86] E. Ilg, Ö. Çiçek, S. Galesso, A. Klein, O. Makansi, F. Hutter, and T. Brox. Uncertainty Estimates and Multi-Hypotheses Networks for Optical Flow. In *European Conference on Computer Vision (ECCV)*, 2018. 4.1, 4.2

[87] S. Im, H. Ha, G. Choe, H.-G. Jeon, K. Joo, and I. Kweon. Accurate 3d reconstruction from small motion clip for rolling shutter cameras. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PP:1–1, 03 2018. 1.3

[88] C. Inoshita, Y. Mukaigawa, Y. Matsushita, and Y. Yagi. Surface Normal Deconvolution: Photometric Stereo for Optically Thick Translucent Objects. In *ECCV*, pages 346–359. 2014. 2.2

[89] H. Ishikawa. Transformation of general binary mrf minimization to the first-order case. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(6):1234–1249, June 2011. 3.4

[90] H. W. Jensen, S. R. Marschner, M. Levoy, and P. Hanrahan. A practical model for subsurface light transport. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '01, pages 511–518, New York, NY, USA, 2001. ACM. 1.3, 1, 5.3, 5.3, 5.6.1, 5.9, 5.10

[91] A. Jones, G. Fyffe, X. Yu, W. C. Ma, J. Busch, R. Ichikari, M. Bolas, and P. Debevec. Head-mounted photometric stereo for performance capture. *Conference for Visual Media Production*\, pages 158–164, 2011. 2.2

[92] J.Y.Bouguet. MATLAB calibration toolbox. `http://www.vision.caltech.edu/bouguetj/calib_doc/`. 2.1, 2.5.1

[93] N. S. Q. C. K. Zhang, J. Xie. Depth sensing beyond lidar range. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 1.3

[94] G. Kamberova and R. Bajcsy. Sensor errors and the uncertainties in stereo reconstruction. In *Empirical Evaluation Techniques in Computer Vision*, pages 96–116. IEEE Computer Society Press, 1998. 4.2

[95] A. Kendall and Y. Gal. What uncertainties do we need in bayesian deep learning for computer vision? In *Advances in Neural Information Processing Systems (NIPS)*, 2017. 4.1, 4.2

[96] A. Kendall, Y. Gal, and R. Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 4.1, 4.2

[97] K. Kim, D. Lee, and I. Essa. Gaussian process regression flow for analysis of motion trajectories. In *International Conference on Computer Vision (ICCV)*, 2011. 4.2

[98] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015. 4.3.3

[99] V. Kolmogorov. A new look at reweighted message passing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(5):919–930, May 2015. 3.4

[100] S. D. Konecky, A. Mazhar, D. Cuccia, A. J. Durkin, J. C. Schotland, and B. J. Tromberg. Quantitative optical tomography of sub-surface heterogeneities using spatially modulated structured light. *Optics Express*, 17(17):14780, aug 2009. 5.2

[101] H. Kubo, S. Jayasuriya, T. Iwaguchi, T. Funatomi, Y. Mukaigawa, and S. G. Narasimhan. Acquiring and characterizing plane-to-ray indirect light transport. In *2018 IEEE International Conference on Computational Photography (ICCP)*, pages 1–10, May 2018. 5.1, 5.4, 5.5, 5.6

[102] W.-S. Lai, J.-B. Huang, O. Wang, E. Shechtman, E. Yumer, and M.-H. Yang. Learning blind video temporal consistency. In *European Conference on Computer Vision (ECCV)*, 2018. 4.4

[103] M. W. Law and A. C. Chung. Three dimensional curvilinear structure detection using optimally oriented flux. *Proceedings of European Conference on Computer Vision (ECCV)*, pages 368–382, 2008. 6.2, 6.3.3, 6.6.4

[104] A. Levin, R. Fergus, F. Durand, and W. T. Freeman. Image and depth from a conventional camera with a coded aperture. *ACM Trans. Graph.*, 26(3):70es, July 2007. 1.2.1

[105] A. Levin, D. Lischinski, and Y. Weiss. A closed form solution to natural image matting. In *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 1*, CVPR '06, pages 61–68, Washington, DC, USA, 2006. IEEE Computer Society. 3.3

[106] J. Liao, B. Buchholz, J. M. Thiery, P. Bauszat, and E. Eisemann. Indoor Scene Reconstruction Using Near-Light Photometric Stereo. *TIP*, 26(3):1089–1101, 2016. 2.1

[107] J. Liao, B. Buchholz, J. M. Thiery, P. Bauszat, and E. Eisemann. Indoor Scene Reconstruction Using Near-Light Photometric Stereo. *IEEE Transactions on Image Processing*, 26(3):1089–1101, 2017. 2.2

[108] T. Lister, P. A. Wright, and P. H. Chappell. Optical properties of human skin. *Journal of Biomedical Optics*, 17(9):1 – 15, 2012. 5.1

[109] B. Liu, S. Gould, and D. Koller. Single image depth estimation from predicted semantic labels. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1253–1260, 2010. 1.3

[110] C. Liu, J. Gu, K. Kim, S. G. Narasimhan, and J. Kautz. Neural rgb-¿d sensing: Depth and uncertainty from a video camera. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10978–10987, 2019. 1.4

[111] C. Liu, A. Maity, A. W. Dubrawski, A. Sabharwal, and S. G. Narasimhan. High resolution diffuse optical tomography using short range indirect subsurface imaging. In *IEEE International Conference*

on Computational Photography, 2020. 1.1, 1.5

[112] C. Liu, A. Maity, A. W. Dubrawski, A. Sabharwal, and S. G. Narasimhan. High resolution diffuse optical tomography using short range indirect subsurface imaging. In *IEEE Conference on Computational Photography (ICCP)*, 2020. 1.4

[113] C. Liu, S. G. Narasimhan, and A. W. Dubrawski. Matting and depth recovery of thin structures using a focal stack. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4782–4790, 2017. 1.4

[114] C. Liu, S. G. Narasimhan, and A. W. Dubrawski. Near-light photometric stereo using circularly placed point light sources. In *IEEE International Conference on Computational Photography (ICCP)*, pages 1–10, 2018. 1.4

[115] F. Liu, C. Shen, and G. Lin. Deep convolutional neural fields for depth estimation from a single image. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015. 1.3

[116] Y. Liu, T. Belkina, J. H. Hays, and R. Lublinerman. Image De-fencing. In *CVPR*, pages 1–8, Jun 2008. 3.2

[117] S. Lynen, T. Sattler, M. Bosse, J. A. Hesch, M. Pollefeys, and R. Siegwart. Get out of my lab: Large-scale, real-time visual-inertial localization. In *Robotics: Science and Systems*, 2015. 1.2.1

[118] W. C. Ma, T. Hawkins, P. Peers, C. F. Chabert, M. Weiss, and P. Debevec. Rapid Acquisition of Specular and Diffuse Normal Maps from Polarized Spherical Gradient Illumination. In *Eurographics Symposium on Rendering (EGSR)*, 2007. 2.2

[119] R. Mahjourian, M. Wicke, and A. Angelova. Unsupervised learning of depth and ego-motion from monocular video using 3D geometric constraints. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 4.2

[120] A. Maimone and H. Fuchs. Reducing interference between multiple structured light depth sensors using motion. In *IEEE Virtual Reality Workshops (VRW)*, pages 51–54, 2012. 4.1, 4.2

[121] I. Matthews and S. Baker. Active appearance models revisited. *International Journal of Computer Vision*, 60(2):135–164, Nov. 2004. 6.2

[122] S. McCloskey. Masking Light Fields to Remove Partial Occlusion. In *ICPR*, pages 2053–2058, 2014. 3.2

[123] M. McGuire, W. Matusik, H. Pfister, J. F. Hughes, and F. Durand. Defocus video matting. *ACM Trans. Graph.*, 24(3):567–576, 2005. 3.3

[124] J. Mei, A. Kirmani, A. Colao, and V. K. Goyal. Phase unwrapping and denoising for time-of-flight imaging using generalized approximate message passing. In *2013 IEEE International Conference on Image Processing*, pages 364–368, 2013. 1.2.2

[125] A. M. Mendonça and A. Campilho. Segmentation of retinal blood vessels by combining the detection of centerlines and morphological reconstruction. *Medical Imaging, IEEE Transactions on*, 25(9):1200–1213, 2006. 6.2

[126] D. Miau, O. Cossairt, and S. Nayar. Focal Sweep Videography with Deformable Optics. In *IEEE International Conference on Computational Photography (ICCP)*, pages 1–8, Apr 2013. 1.2.1

[127] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *arXiv*, 2020. 1.2.1

[128] R. Mur-Artal and J. D. Tardós. ORB-SLAM2: an open-source SLAM system for monocular, stereo and RGB-D cameras. *IEEE Transactions on Robotics*, 33(5):1255–1262, 2017. 4.2

[129] Z. Murez, T. Treibitz, R. Ramamoorthi, and D. Kriegman. Photometric stereo in a scattering medium. *Proceedings of the IEEE International Conference on Computer Vision*, (October):3415–3423, 2015. 2.2

[130] S. G. Narasimhan and S. K. Nayar. Chromatic framework for vision in bad weather. In *Proceedings*

IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No.PR00662), volume 1, pages 598–605 vol.1, 2000. 1.3

[131] P. K. Nathan Silberman, Derek Hoiem and R. Fergus. Indoor segmentation and support inference from RGBD images. In *European Conference on Computer Vision (ECCV)*, 2012. 1.2.1, 4.4

[132] S. Nayar, G. Krishnan, M. D. Grossberg, and R. Raskar. Fast Separation of Direct and Global Components of a Scene using High Frequency Illumination. *ACM Trans. on Graphics (also Proc. of ACM SIGGRAPH)*, Jul 2006. 1.2.2

[133] S. K. Nayar, K. Ikeuchi, and T. Kanade. Shape from interreflections. *International Journal of Computer Vision*, 6(3):173–195, 1991. 2.2

[134] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohli, J. Shotton, S. Hodges, and A. Fitzgibbon. KinectFusion: Real-time dense surface mapping and tracking. In *IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 127–136, 2011. 4.1, 4.1, 4.1, 4.2, 4.4

[135] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison. Dtam: Dense tracking and mapping in real-time. In *2011 International Conference on Computer Vision*, pages 2320–2327, Nov 2011. 1, 1.2.1

[136] R. Ng, M. Levoy, M. Bredif, G. Duval, M. Horowitz, and P. Hanrahan. Light Field Photography with a Hand-held Plenoptic Camera. pages 1–11, Apr 2005. 3.6.2

[137] M. Nießner, M. Zollhöfer, S. Izadi, and M. Stamminger. Real-time 3D reconstruction at scale using voxel hashing. *ACM Transactions on Graphics (TOG)*, 2013. 4.1, 4.1, 4.1, 4.4, 4.4, 4.4, 4.11, 4.13

[138] D. Nordsletten, S. Blackett, M. Bentley, E. Ritman, and N. Smith. Structural morphology of renal vasculature. *American Journal of Physiology - Endocrinology and Metabolism*, 291(1), 2006. 3.1

[139] H. Obrig and A. Villringer. Beyond the visible—imaging the human brain with light. *Journal of Cerebral Blood Flow & Metabolism*, 23(1):1–18, jan 2003. 5.1

[140] M. A. O'Leary. *Imaging with diffuse photon density waves*. PhD thesis, University of Pennsylvania, 1996. 5.2

[141] M. O'Toole, S. Achar, S. G. Narasimhan, and K. N. Kutulakos. Homogeneous codes for energy-efficient illumination and imaging. *ACM Trans. Graph.*, 34(4):35:1–35:13, July 2015. 5.1

[142] M. OToole, S. Achar, S. G. Narasimhan, and K. N. Kutulakos. Homogeneous codes for energy-efficient illumination and imaging. *ACM Trans. Graph.*, 34(4), July 2015. 1.2.2

[143] T. Papadhimitri and P. Favaro. Uncalibrated Near-Light Photometric Stereo. In *BMVC*, pages 1–12, 2014. 2.2

[144] J. J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 1.2.1, 7.1

[145] V. Pătrăucean, P. Gurdjos, and R. G. von Gioi. A parameterless line segment and elliptical arc detector with enhanced ellipse fitting. In *ECCV 2012*. 2.6, 2.5.1

[146] F. Pittaluga, Z. Tasneem, J. Folden, B. Tilmon, A. Chakrabarti, and S. J. Koppal. A mems-based foveating lidar to enable real-time adaptive depth sensing. In *arXiv*, 2020. 7.1

[147] F. Pomerleau, A. Breitenmoser, M. Liu, F. Colas, and R. Siegwart. Noise characterization of depth sensors for surface inspections. In *International Conference on Applied Robotics for the Power Industry (CARPI)*, pages 16–21, 2012. 4.2

[148] M. W. Powell, S. Sarkar, and D. Goldgof. A simple strategy for calibrating the geometry of light sources. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(9):1022–1027, 2001. 2.1, 2.5.1

[149] A. Puszka, L. D. Sieno, A. D. Mora, A. Pifferi, D. Contini, A. Planat-Chrétien, A. Koenig, G. Boso, A. Tosi, L. Hervé, and J.-M. Dinten. Spatial resolution in depth for time-resolved diffuse optical tomography using short source-detector separations. *Biomedical Optics Express*, 6(1):1, dec 2014.

5.2

[150] C. R. Qi, W. Liu, C. Wu, H. Su, and L. J. Guibas. Frustum PointNets for 3D object detection from RGB-D data. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 4.1, 4.2

[151] Y. Quéau, B. Durix, T. Wu, D. Cremers, F. Lauze, and J.-D. Durou. Led-based photometric stereo: Modeling, calibration and numerical solution. *Journal of Mathematical Imaging and Vision*, Sep 2017. 1.2.2, 1.4, 2.2, 2.7, 2.9, 2.6.2, 2.6.3

[152] M. D. Reisman, Z. E. Markow, A. Q. Bauer, and J. P. Culver. Structured illumination diffuse optical tomography for noninvasive functional neuroimaging in mice. *Neurophotonics*, 4(2):021102, apr 2017. 5.2

[153] F. Remondino and D. Stoppa. *TOF Range-Imaging Cameras*. Springer Publishing Company, Incorporated, 2013. 4.1, 4.2

[154] M. Reynolds, J. Dobo, L. Peel, T. Weyrich, and G. J. Brostow. Capturing time-of-flight data with confidence. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011. 4.2

[155] H. Santo, M. Samejima, Y. Sugano, B. Shi, and Y. Matsushita. Deep photometric stereo network. In *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, pages 501–509, 2017. 1.2.2

[156] V. Saragadam and A. C. Sankaranaryanan. Programmable Spectrometry: Per-pixel material classification using learned spectral filters. In *IEEE Intl. Conf. Computational Photography (ICCP)*, 2020. 7.1

[157] G. Satat, C. Barsi, and R. Raskar. Skin perfusion photography. In *Proceedings of IEEE International Conference on Computational Photography (ICCP)*, pages 1–8, 2014. 6.2

[158] A. Saxena, S. H. Chung, and A. Y. Ng. 3D depth reconstruction from a single still image. *International Journal of Computer Vision (IJCV)*, 76(1):53–69, Jan. 2008. 4.1, 4.2

[159] A. Saxena, J. Schulte, and A. Y. Ng. Depth estimation using monocular and stereo cues. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, IJCAI'07, pages 2197–2203, 2007. 4.2

[160] B. Schoenemann, H. Pärnaste, and E. N. K. Clarkson. Structure and function of a compound eye, more than half a billion years old. *Proceedings of the National Academy of Sciences*, 114(51):13489–13494, 2017. 1

[161] J. L. Schönberger and J.-M. Frahm. Structure-from-motion revisited. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 4.2

[162] J. L. Schönberger, E. Zheng, M. Pollefeys, and J.-M. Frahm. Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision (ECCV)*, 2016. 1.2.1, 4.2

[163] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 519–528, June 2006. 1.2.1

[164] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006. 4.1, 4.2

[165] J. Selb, A. M. Dale, and D. A. Boas. Linear 3d reconstruction of time-domain diffuse optical imaging differential data: improved depth localization and lateral resolution. *Optics Express*, 15(25):16400, 2007. 5.2

[166] B. Shi, Z. Wu, Z. Mo, D. Duan, S.-K. Yeung, and P. Tan. A Benchmark Dataset and Evaluation for Non-Lambertian and Uncalibrated Photometric Stereo. *CVPR*, pages 3707–3716, 2016. 2.2

[167] T. Shimokawa, T. Ishii, Y. Takahashi, S. Sugawara, M. aki Sato, and O. Yamashita. Diffuse optical tomography using multi-directional sources and detectors. *Biomedical Optics Express*, 7(7):2623,

jun 2016. 1.2.2, 5.2

[168] J. Shotton, B. Glocker, C. Zach, S. Izadi, A. Criminisi, and A. Fitzgibbon. Scene coordinate regression forests for camera relocalization in RGB-D images. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013. 1.2.1, 4.4, 4.1, 4.4

[169] A. Siegel, J. J. Marota, and D. Boas. Design and evaluation of a continuous-wave diffuse optical tomography system. *Optics Express*, 4(8):287, apr 1999. 1.2.2, 5.2

[170] A. Sironi, V. Lepetit, and P. Fua. Multiscale Centerline Detection by Learning a Scale-Space Distance Transform. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2697–2704, 2014. 6.2, 6.3.3

[171] V. Sitzmann and G. Zollhofer, Michaeland Wetzstein. Scene representation networks: Continuous 3d-structure-aware neural scene representations. In *NeurIPS*, 2019. 1.2.1, 7.1

[172] W. A. P. Smith, R. Ramamoorthi, and S. Tozza. Linear depth estimation from an uncalibrated, monocular polarisation image. In B. Leibe, J. Matas, N. Sebe, and M. Welling, editors, *ECCV*, 2016. 1.3, 7.1

[173] S. Song, S. P. Lichtenberg, and J. Xiao. SUN RGB-D: A RGB-D scene understanding benchmark suite. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 4.1

[174] C. Steger. An Unbiased Detector of Curvilinear Structures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–13, Feb. 1998. 6.2, 6.3.2

[175] G. A. Strasser, J. S. Kaminker, and M. Tessier-Lavigne. Microarray analysis of retinal endothelial tip cells identifies cxcr4 as a mediator of tip cell morphology and branching. *Blood*, 115(24):5102–5110, 2010. 3.1

[176] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers. A benchmark for the evaluation of RGB-D SLAM systems. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2012. 4.4

[177] D. Sun, S. Roth, and M. J. Black. Secrets of Optical Flow Estimation and Their Principles. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, Mar. 2010. 6.2, 6.4, 6.5

[178] S. Suwajanakorn, C. Hernandez, and S. M. Seitz. Depth from focus with your mobile phone. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015. 3.1

[179] R. Szeliski. Bayesian modeling of uncertainty in low-level vision. *International Journal of Computer Vision*, 5(3):271–301, Dec 1990. 4.2

[180] M. W. Tao, J. Su, T. Wang, J. Malik, and R. Ramamoorthi. Depth estimation and specular removal for glossy surfaces using point and line consistency with light-field cameras. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(6):1155–1169, 2016. 1.3

[181] D. Tarlow, R. S. Zemel, and I. E. Givoni. HOP-MAP : Efficient Message Passing with High Order Potentials. *Journal of Machine Learning Research*, pages 812–819, 2010. 3.4

[182] K. Tateno, F. Tombari, I. Laina, and N. Navab. CNN-SLAM: Real-time dense monocular SLAM with learned depth prediction. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 4.1, 4.2

[183] M. Thomas, E. Laude, M. Moeller, J. Lellmann, and D. Cremers. SublabelAccurate Relaxation of Nonconvex Energies. In *CVPR*, number 1, pages 3948–3956, 2016. 3.6.4, 3.1, 3.6.4

[184] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press, 2005. 1.2.1

[185] Y. Tian and S. G. Narasimhan. Globally Optimal Estimation of Nonrigid Image Distortion. *International Journal of Computer Vision*, 98(3):279–302, July 2012. 6.2

[186] B. Tippetts, D. J. Lee, K. Lillywhite, and J. Archibald. Review of stereo vision algorithms and their suitability for resource-limited systems. *Journal of Real-Time Image Processing*, 11(1):5–25, 2016.

4.1

[187] Tomoaki Higo, Yasuyuki Matsushita, Neel Joshi, and Katsushi Ikeuchi. A hand-held photometric stereo camera for 3-D modeling. *ICCV*, (Iccv):1234–1241, 2009. 2.2

[188] X. Tong, J. Wang, J. Wang, S. Lin, B. Guo, H.-Y. Shum, and H.-Y. Shum. Modeling and rendering of quasi-homogeneous materials. *ACM Trans. Graph.*, 24(3):1054–1061, July 2005. 1

[189] S. Trachsel, S. M. Kaeppler, K. M. Brown, and J. P. Lynch. Shovelomics: high throughput phenotyping of maize (zea mays l.) root architecture in the field. *Plant and Soil*, 341(1):75–87, 2011. 3.1

[190] B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon. Bundle adjustment a modern synthesis. In *International Conference on Computer Vision (ICCV)*, 1999. 4.3.4

[191] J. Tuley, N. Vandapel, and M. Hebert. Analysis and removal of artifacts in 3-d LIDAR data. In *International Conference on Robotics and Automation (ICRA)*, 2005. 4.2

[192] E. Turetken, C. Becker, P. Glowacki, F. Benmansour, and P. Fua. Detecting Irregular Curvilinear Structures in Gray Scale and Color Imagery Using Multi-directional Oriented Flux. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pages 1553–1560. IEEE, Nov. 2013. 6.2

[193] B. Ummenhofer, H. Zhou, J. Uhrig, N. Mayer, E. Ilg, A. Dosovitskiy, and T. Brox. DeMoN: Depth and motion network for learning monocular stereo. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 4.1, 4.1, 4.2, 4.4, 4.1, 4.11, 4.12

[194] V. Vaish, M. Levoy, R. Szeliski, C. L. Zitnick, and S. B. Kang. Reconstructing Occluded Surfaces Using Synthetic Apertures: Stereo, Focus and Robust Measures. In *CVPR*, pages 2331–2338, 2006. 3.2, 3.6.3, 3.8

[195] R. Van Uitert and I. Bitter. Subvoxel precise skeletons of volumetric data based on fast marching methods. *Medical Physics*, 34(2):627, 2007. 6.2

[196] A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms. `http://www.vlfeat.org/`, 2008. 3.5

[197] A. Villringer. Non-invasive optical spectroscopy and imaging of human brain function. *Trends in Neurosciences*, 20(10):435–442, oct 1997. 5.1

[198] N. Wadhwa, M. Rubinstein, F. Durand, and W. T. Freeman. Riesz pyramids for fast phase-based video magnification. In *Proceedings of IEEE International Conference on Computational Photography (ICCP)*, pages 1–10, 2014. 6.2

[199] C. Wang, J. M. Buenaposada, R. Zhu, and S. Lucey. Learning depth from monocular videos using direct methods. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 1.2.1, 4.1, 4.2

[200] T.-C. Wang, M. Chandraker, A. A. Efros, and R. Ramamoorthi. SVBRDF-Invariant Shape and Reflectance Estimation from Light-Field Cameras. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (March):5451–5459, 2016. 1

[201] T.-c. Wang, A. A. Efros, and R. Ramamoorthi. Occlusion-aware Depth Estimation Using Light-field Cameras. In *ICCV*, 2015. 2.2, 3.2, 3.6.4, 3.5, 3.6

[202] M. Watanabe and S. Nayar. Telecentric Optics for Focus Analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(12):1360–1365, Dec 1997. 1.2.1

[203] M. Watanabe and S. Nayar. Rational Filters for Passive Depth from Defocus. *International Journal on Computer Vision*, 27(3):203–225, May 1998. 1.2.1

[204] T. Whelan, S. Leutenegger, R. S. Moreno, B. Glocker, and A. Davison. ElasticFusion: dense SLAM without a pose graph. In *Robotics: Science and Systems (RSS)*, 2015. 4.1, 4.2

[205] R. J. Woodham. Photometric method for determining surface orientation from multiple images, 1980. 2.2

[206] C. Wu, S. G. Narasimhan, and B. Jaramaz. A multi-image shape-from-shading framework for near-lighting perspective endoscopes. *International Journal of Computer Vision*, 86(2-3):211–228, 2010. 2.2

[207] H.-Y. Wu, M. Rubinstein, E. Shih, J. Guttag, F. Durand, and W. Freeman. Eulerian video magnification for revealing subtle changes in the world. *ACM Transactions on Graphics (TOG)*, 31(4):1–8, July 2012. 6.2, 6.4

[208] W. Xie, C. Dai, and C. C. L. Wang. Photometric stereo with near point lighting: A solution by mesh deformation. In *CVPR*, volume 07-12-June, pages 4585–4593, 2015. 2.2

[209] Y. Xiong, A. Chakrabarti, R. Basri, S. J. Gortler, D. W. Jacobs, and T. Zickler. From shading to local shape. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(1):67–79, 2015. 2.6.3

[210] N. Xu, B. Price, S. Cohen, and T. Huang. Deep image matting. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 311–320, 2017. 7.1

[211] T. Xue, M. Rubinstein, C. Liu, and W. T. Freeman. A computational approach for obstruction-free photography. *ACM Transactions on Graphics (TOG)*, 34(4):79:1–79:11, jul 2015. 1.3, 3.1, 3.2

[212] S. Yamazaki, S. G. Narasimhan, S. Baker, and T. Kanade. The theory and practice of coplanar shadowgram imaging for acquiring visual hulls of intricate objects. *IJCV*, 81(3):259–280, 2009. 1.3, 3.2

[213] F. Yang, F. Gao, P. Ruan, and H. Zhao. Combined domain-decomposition and matrix-decomposition scheme for large-scale diffuse optical tomography. *Applied Optics*, 49(16):3111, may 2010. 5.2

[214] Y. Yao, Z. Luo, S. Li, T. Fang, and L. Quan. MVSNet: Depth inference for unstructured multi-view stereo. In *European Conference on Computer Vision (ECCV)*, 2018. 1.2.1, 4.1, 4.2

[215] X. Yi, W. Chen, L. Wu, W. Zhang, J. Li, X. Wang, L. Zhang, H. Zhao, and F. Gao. Towards diffuse optical tomography of arbitrarily heterogeneous turbid medium using GPU-accelerated monte-carlo forward calculation. In F. S. Azar and X. Intes, editors, *Multimodal Biomedical Imaging VIII*. SPIE, mar 2013. 5.2

[216] Z. Yin and J. Shi. GeoNet: Unsupervised learning of dense depth, optical flow and camera pose. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 4.2

[217] F. Yu and D. Gallup. 3d reconstruction from accidental motion. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3986–3993, 2014. 1.3

[218] C. Zhang, L. Wu, C. Zheng, I. Gkioulekas, R. Ramamoorthi, and S. Zhao. A differential theory of radiative transfer. *ACM Trans. Graph.*, 38(6), 2019. 1.3, 5.2

[219] X. Zhang. Instrumentation in diffuse optical imaging. *Photonics*, 1(1):9–32, mar 2014. 1.2.2, 5.2

[220] H. Zhou, B. Ummenhofer, and T. Brox. DeepTAM: Deep tracking and mapping. In *European Conference on Computer Vision (ECCV)*, 2018. 1, 1.2.1, 4.1, 4.2

[221] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe. Unsupervised learning of depth and ego-motion from video. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1.2.1, 4.1, 4.2

[222] Z. Zhou and P. Tan. Ring-light Photometric Stereo. In *ECCV*, 2010. 2.2

[223] H. zgr Kazanci, T. Mercan, and M. Canpolat. Design and evaluation of a reflectance diffuse optical tomography system. *Optical and Quantum Electronics*, 47(2):257–265, mar 2014. 1.3