High Quality Structured Light 3D Scanning with Low Calibration Effort

Utkarsh

High Quality Structured Light 3D Scanning with Low Calibration Effort

Project report submitted to

Indian Institute of Technology Kharagpur

for the award of the degree

of

Master of Technology

in Electrical Engineering

with specialization Instrumentation and Signal Processing

by

Utkarsh

Department of Electrical Engineering

Indian Institute of Technology Kharagpur

April 2019

DECLARATION

I certify that

(a) The work contained in this project report is original and has been done by me under the guidance of my supervisor.

(b) The work has not been submitted to any other Institute for any degree or diploma.

(c) I have followed the guidelines provided by the institute in preparing the project report.

(d) I have conformed to the norms and guidelines given in the Ethical Code of Conduct of the Institute.

(e) Whenever I have used materials (data, theoretical analysis, figures, and text) from other sources, I have given due credit to them by citing them in the text of the thesis and giving their details in the references. Further, I have taken permission from the copyright owners of the sources, whenever necessary.

Date: April 2019 Utkarsh

Place: Kharagpur 17EE64R11

DEPARTMENT OF ELECTRICAL ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY KHARAGPUR

KHARAGPUR - 721302, INDIA

CERTIFICATE

This is to certify that the project report entitled "**High Quality Structured Light 3D Scanning with Low Calibration Effort**" submitted by **Utkarsh** (Roll No. 17EE64R11) to Indian Institute of Technology Kharagpur is a record of bona fide research work under my supervision and is worthy of consideration for the award of the degree of Master of Technology in Instrumentation and Signal Processing of the Institute.

Date: April 2019
Place: Kharagpur

Dr. Avishek Chatterjee
Department of Electrical Engineering
Indian Institute of Technology Kharagpur
Kharagpur - 721302, India

Acknowledgements

I would like to take this opportunity to express my deep gratitude for the supervision and guidance provided by my supervisor, Dr. Avishek Chatterjee. He has always pushed me to think critically and took out time from his busy schedule for all the discussions. It has been a great learning experience working under his supervision.

This work was supported by the 3D Heritage Digitization project (project code: 3HD) funded by the ISIRD, SRIC, Indian Institute of Technology, Kharagpur. I would also like to thank the Department of Electrical Engineering, Indian Institute of Technology, Kharagpur for making available the laboratory space necessary to undertake this project.

Abstract

Structured Light systems have been in use for about two decades now. They have become quite popular recently as the prices of high quality digital cameras and projectors have gone down. Structured Light 3D scanners can be assembled relatively easily using parts available off the shelf. The most important and tedious step in setting up a structured light 3D scanner is its calibration. Inaccurate calibration can result in distorted scans. Several widely used methods exist for calibration of structured light systems. Most methods calibrate the system prior to scanning and require extra hardware such as calibration rigs or boards. As such, these methods are time consuming and tedious. Calibration is required every time the relative position of the devices is changed. This necessitates an on the go calibration method which calibrates the system during scanning itself and does not require any special boards or rigs .

In this project, it has been proposed to use a calibrated RGBD camera for the geometric calibration of the structured light system. A RGBD camera and a high resolution digital camera along with a projector are used to get high quality 3D scans with low calibration effort. It is also proposed to use the two cameras to minimize occlusion while ensuring that the 3D points are well triangulated by fusing the data from small and large baseline setups. The effects of radiometric non-linearity have been studied and discussed. The reconstruction quality achieved with the proposed method is much superior to the scans achieved with commercial depth scanners like Microsoft Kinect which do not require calibration either and has been used in the proposed setup.

Keywords – Geometric Calibration, Structured light, Radiometric Estimation, RGBD Camera, Occlusion

Contents

List of Figures

List of Tables

Abbreviations

SLS	**S**tructured **L**ight **S**canner/**S**ystem
DLT	**D**irect **L**inear **T**ransformation
SVD	**S**ingular **V**alue **D**ecomposition

Symbols

P	Projection Matrix
K	Intrinsic Calibration Matrix
M	Extrinsic Calibration Matrix
R	Rotation Matrix
t	Translation Vector
X	3D coordinates of a point
x	Pixel coordinates on the image plane

Chapter 1

Introduction

Structured Light 3D Scanners (SLS) are used to capture the shape, size and structure of objects. 3D scanners have various applications such as in the fields of movie production, orthotics / prosthetics, digitization of artifacts, etc. With the lowering prices of cameras and projectors, structured light scanners can be built with devices available directly off the shelf.

SLS capture high quality, dense and accurate scans. They work on the principle of stereo geometry, that is, two views are used to triangulate the depth. These systems can be set up by replacing one of the cameras in a stereo camera pair by a projector. In camera stereo pairs, finding correspondences is computationally expensive and some times even impossible. This problem is mitigated in structured light systems by projecting patterns to encode the view space.

This chapter is dedicated to the understanding of geometry behind the scanning process and why structured light is required. The objectives of this project and the organization of this report are described as well.

1.1 Camera Geometry

Structured light 3D scanners are built using a camera and a projector. To understand how SLS works, one must understand camera geometry and how an image is formed. Figure 1.1 depicts a pinhole camera viewing an object. The pinhole camera is a simplified camera model which consists of a photographic film or sensor inside a box having a small hole (the aperture) at the opposite end. Without the box, any point on the film will correspond to multiple directions. When the film is enclosed within a box with a small aperture, any point on the film has only a single ray direction associated with it. If the aperture is small, the image formed is sharp but dark. As the size of the aperture is increased, the image brightness increases at the cost of sharpness.

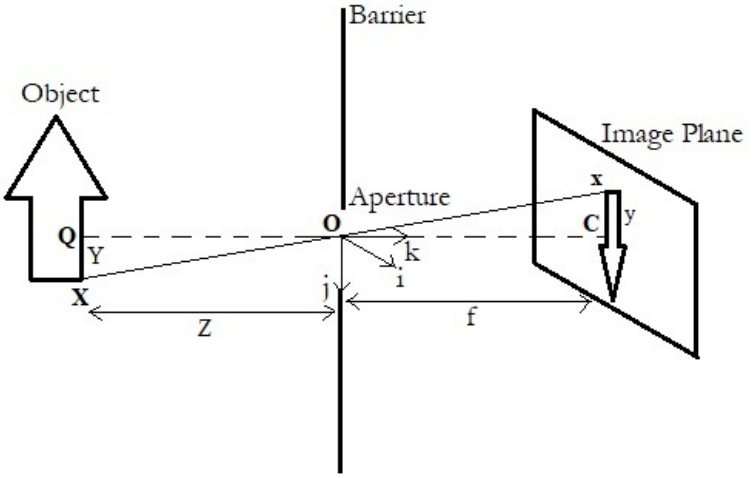


FIGURE 1.1: Pinhole Camera Model

For all further discussions, the film is referred to as the image plane. The aperture is denoted by the point **O** and is also known as the center of projection, the camera center or the optical center. A coordinate system [i j k] is attached to the pinhole **O** and is defined such that the axis k is perpendicular to the image plane and points towards it. This axis is called the optical axis of the camera system, and the coordinate system is known as the camera reference system or camera coordinate

system. **C** is the intersection of the optical axis with the image plane and is known as the principal point or the image center. The perpendicular distance between the aperture and the image plane is the focal length f.

As can be seen in Figure 1.1, the image of point **X** on the object is formed at the point **x** in the image plane. From the two similar triangles **OQX** and **OCx**, a relationship between the image coordinates and the world coordinates can be derived as

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} f\frac{X}{Z} \\ f\frac{Y}{Z} \end{bmatrix} \tag{1.1}$$

where $\mathbf{X} = [X, Y, Z]^T$ and $\mathbf{x} = [x, y]^T$.

1.2 The Digital Image Space

The mapping of a point in 3D space to its image in the image plane is known as projective transformation. This projection of 3D points in the image plane is not what is actually seen in digital images for several reasons. First, points in the digital images are, in general, in a different reference system than those in the image plane. Second, digital images are divided into discrete pixels, whereas points in the image plane are continuous. Finally, the physical sensors can introduce non-linearity such as distortion and skew to the mapping. To account for these differences, a number of additional transformations are required which facilitate in mapping any point from the 3D world to pixel coordinates.

Image coordinates have their origin **C** at the image center where the k axis intersects the image plane. On the other hand, digital images typically have their origin at the lower-left corner of the image. Thus, 2D points in the image plane and 2D points in the image are offset by a translation vector $[c_x, c_y]^T$. To accommodate this change

of coordinate systems, the mapping now becomes:

$$\mathbf{x} = \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} f\frac{X}{Z} + c_x \\ f\frac{Y}{Z} + c_y \end{bmatrix} \tag{1.2}$$

Next, the points in digital images are expressed in pixels, while the points in the image plane are represented in physical measurements (e.g. millimeters). In order to accommodate this change of units, two parameters k and l are used. These parameters, whose units are something like $\frac{pixels}{mm}$, correspond to the change of units in the two axes of the image plane. k and l may be different because the aspect ratio of the unit element is not always one. If $k = l$, the camera has square pixels. The previous mapping is now adjusted to be:

$$\mathbf{x} = \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} fk\frac{X}{Z} + c_x \\ fl\frac{Y}{Z} + c_y \end{bmatrix} = \begin{bmatrix} \alpha\frac{X}{Z} + c_x \\ \beta\frac{Y}{Z} + c_y \end{bmatrix} \tag{1.3}$$

It is clear from the above equation that the projection $\mathbf{X} \rightarrow \mathbf{x}$ is not linear, as the operation divides one of the input parameters (Z). To linearize these equations, a new coordinate is used such that the point $\mathbf{x} = (x, y)$ becomes $(x, y, 1)$. Similarly, the 3D point $\mathbf{X} = (X, Y, Z)$ becomes $(X, Y, Z, 1)$. This is referred to as the homogeneous coordinate system. Using homogeneous coordinates, the equations become:

$$\mathbf{x}_h = \begin{bmatrix} \alpha X + c_x Z \\ \beta Y + c_y Z \\ Z \end{bmatrix} = \begin{bmatrix} \alpha & 0 & c_x & 0 \\ 0 & \beta & c_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha & 0 & c_x & 0 \\ 0 & \beta & c_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \mathbf{X}_h \tag{1.4}$$

For further discussions, the index h will not be used and any point coordinates $\mathbf{x}$ or $\mathbf{X}$ should be assumed to be homogeneous unless stated otherwise. The relationship between the 3D coordinates of a point and its image coordinates can now be written

as:

$$\mathbf{x} = \begin{bmatrix} \alpha & 0 & c_x & 0 \\ 0 & \beta & c_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha & 0 & c_x \\ 0 & \beta & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{0} \end{bmatrix} \mathbf{X} = \mathbf{K} \begin{bmatrix} \mathbf{I} & \mathbf{0} \end{bmatrix} \mathbf{X} \tag{1.5}$$

The matrix $\mathbf{K}$ is known as the **intrinsic calibration matrix** and contains some of the critical parameters that are useful to characterize a camera model. There are two more parameters that may be present in cameras: **skewness** and **distortion**. An image is said to be skewed when the camera coordinate system is skewed, that is, the angle between the two axes is slightly larger or smaller than 90 degrees. The new intrinsic calibration matrix accounting for skewness (γ) is:

$$\mathbf{K} = \begin{bmatrix} \alpha & \gamma & c_x \\ 0 & \beta & c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{1.6}$$

So far the point $\mathbf{X}$ has been assumed to be in the 3D camera reference system. But, it may be so that the information about the 3D world is available in a different coordinate system. Thus, an additional transformation that relates points from the world reference system to the camera reference systems is required. This transformation is captured by a rotation matrix $\mathbf{R}$ and a translation vector $\mathbf{t}$. These are known as **extrinsic calibration parameters**. Therefore, given a point in a world reference system $\mathbf{X}_w$, its camera coordinates can be computed as:

$$\mathbf{X} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{bmatrix} \mathbf{X}_w \tag{1.7}$$

Substituting this into equation 1.5 and simplifying gives

$$\mathbf{x} = \mathbf{K} \begin{bmatrix} \mathbf{R} & \mathbf{t} \end{bmatrix} \mathbf{X}_w = \mathbf{K}\mathbf{M}\mathbf{X}_w = \mathbf{P}\mathbf{X}_w \tag{1.8}$$

This completes the mapping from a 3D point $\mathbf{X}_w$ in an arbitrary world reference

system to the image plane. The projection matrix **P** consists of two types of parameters: **intrinsic** (matrix **K**) and **extrinsic** (matrix **M**). Parameters contained in the matrix **K** are the intrinsic parameters, which change as the camera changes. The extrinsic parameters include the rotation and translation, which do not depend on the camera's build. Overall, the 3×4 projection matrix **P** has 11 degrees of freedom: 5 from the intrinsic matrix, 3 from extrinsic rotation, and 3 from extrinsic translation. Since the homogeneous coordinates of **x** only give a ray direction, the mapping in equation 1.8 holds only up to a scale. The mapping is better expressed as $\mathbf{x} \propto \mathbf{P}\mathbf{X}_w$.

1.3 The Stereo Camera Pair

The depth of a scene cannot be captured using a single camera as every point on the image only gives a ray direction associated with it. Thus, it cannot be known which point on that ray actually formed the image. This problem is easily resolved by using two cameras separated by some distance viewing the same object.

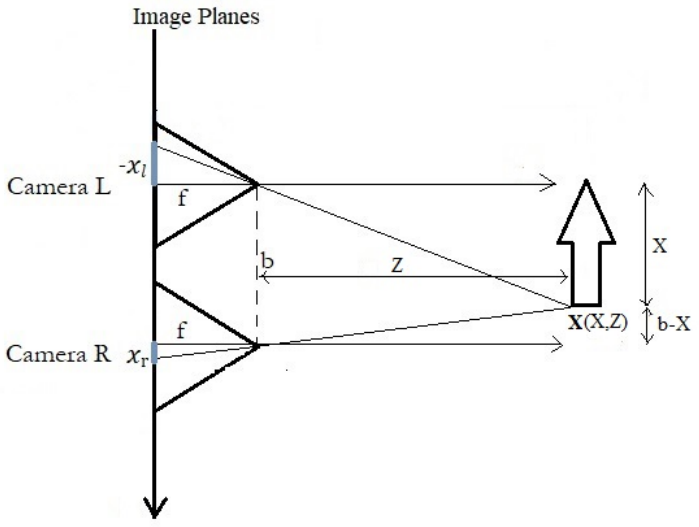


FIGURE 1.2: Stereo Camera Pair

As can be seen in Figure 1.2, two cameras (camera L and camera R) are viewing the same object. Their optical centres are separated by a distance b known as the baseline. Both cameras have a focal length f. The image of point $\mathbf{X}$ on the object is formed at x coordinates x_l and x_r on the cameras L and R respectively. From equation 1.1, the relationship between $\mathbf{X}$ and its projections are:

$$\frac{Z}{f} = \frac{X}{-x_l} \tag{1.9}$$

$$\frac{Z}{f} = \frac{b - X}{x_r} \tag{1.10}$$

From the above two equations, the Z coordinate of point $\mathbf{X}$ is:

$$Z = \frac{f \times b}{x_r - x_l} = \frac{f \times b}{d} \tag{1.11}$$

where d is known as disparity. This method of estimating 3D coordinates of a point is known as stereo triangulation.

Scanning using stereo camera pairs poses two main problems:

- Correspondence Matching
- Calibration

These problems are discussed in brief in the next few paragraphs.

Correspondence Matching

Figure 1.3 shows images captured from two cameras forming a stereo pair. For triangulating the depth of any point, its image should be formed on the image planes of both the cameras. The next problem is to find which two pixels in the two images correspond to the same 3D location. This is computationally demanding and

FIGURE 1.3: Images from left and right camera views respectively

requires that the objects have some features that could be matched. For example, in figure 1.3, the objects such as the ball, the box, and the idol have some texture and edges which can be matched. But, the wall behind does not have any texture and it is impossible to accurately find which two pixels in the two images correspond to the same point on the wall. Due to this, the scans from stereo camera pairs are sparse and have high computational requirements. This issue is easily addressed in structured light systems and will be discussed in more detail in section 1.4.

Calibration

In any measurement system, calibration is the most important part for accurate measurements. Same is the case for 3D scanners. While discussing the stereo camera pair, in figure 1.2, it was assumed that both the cameras are identical, that they are separated only along a single axis, and that their image planes lie on the same plane. This is a highly simplified scenario and is rarely possible in practice.

Figure 1.4 shows a more practical stereo arrangement. Here, the cameras are not identical. Their intrinsic parameters ($\mathbf{K}_1$ and $\mathbf{K}_2$) are different. Their image planes do not lie on the same plane. They are translated as well as rotated with respect

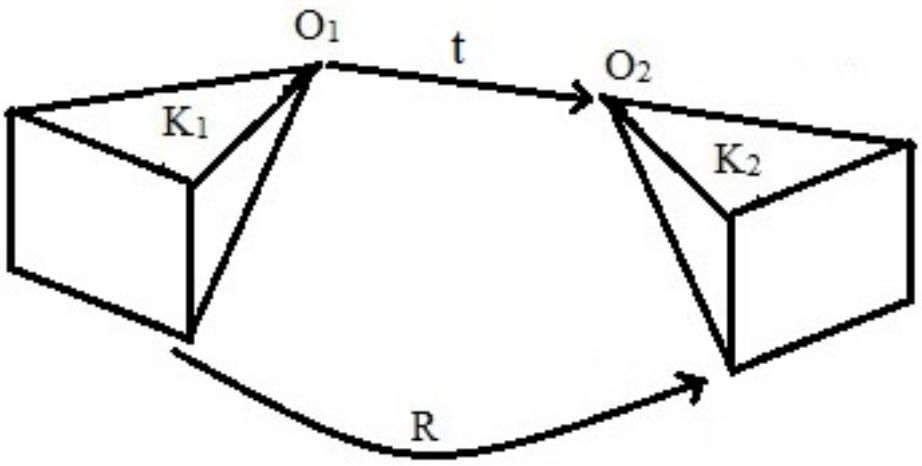


FIGURE 1.4: A generic stereo pair arrangement

to each other. For accurate scanning, $\mathbf{K}_1$, $\mathbf{K}_2$, $\mathbf{R}$, and $\mathbf{t}$ should be known precisely. Estimating these parameters is known as calibration. More discussion on this will follow in upcoming sections.

1.4 Structured Light Scanners

As discussed earlier, a significant problem in stereo camera pairs is that of correspondence matching. This problem is resolved in SLS by replacing one of the cameras of the camera stereo pair by a projector or by adding a projector to the camera stereo pair. As can be seen in figure 1.5, a pattern is projected on to the objects being scanned. As viewed from the camera, the pattern is deformed due to the object shape. By measuring these deformities in the patterns, the shape, size, and structure of the objects can be measured.

The purpose of the patterns is to encode each pixel in the image uniquely so that the correspondence matching problem becomes trivial. There are quite a few different coding strategies in use today. Salvi et al. [2004] have elaborately discussed many of

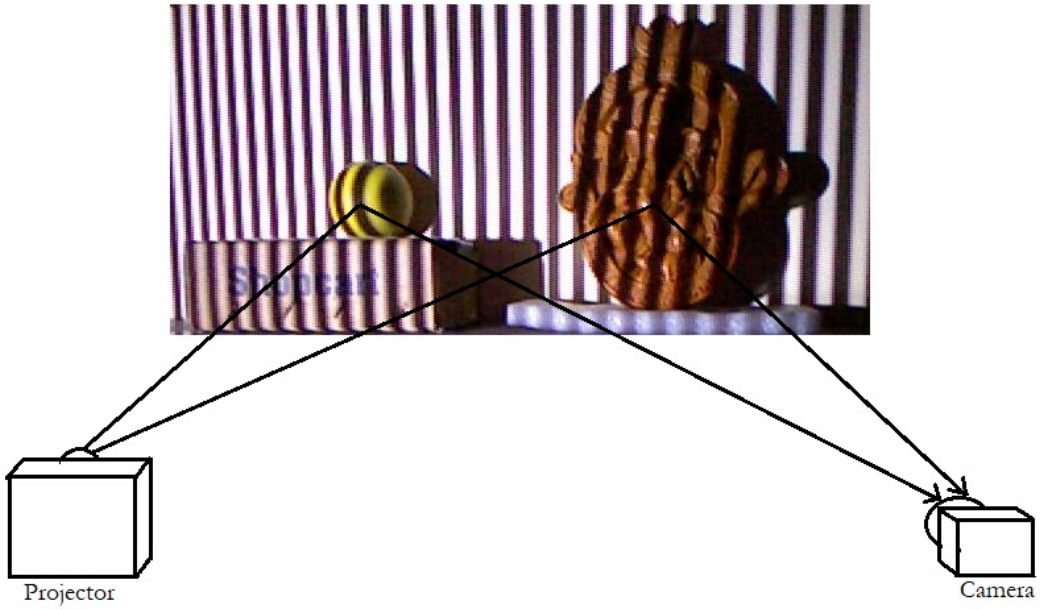


FIGURE 1.5: Structured Light System Setup

the coding strategies that have been developed for SLS along with the advantages and disadvantages of each.

Binary coded pattern (figure 1.6) is one of the coding schemes used in SLS. A number of patterns are projected over time so as to encode the pixels along the horizontal direction uniquely. Figure 1.7 shows how the codes are spatially distributed when five binary coded patterns are projected over time.

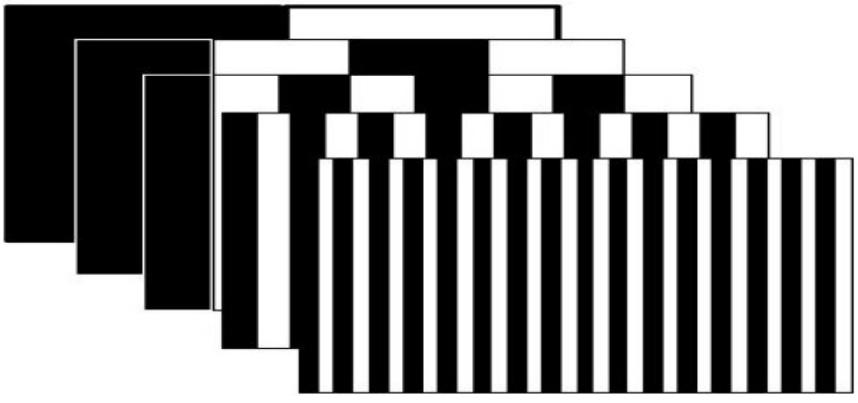


FIGURE 1.6: A sequence of projected binary patterns (Geng [2011])

The code at a given camera pixel is the x coordinate of the corresponding projector pixel. Thus, the 3D points can be triangulated with the known correspondences.

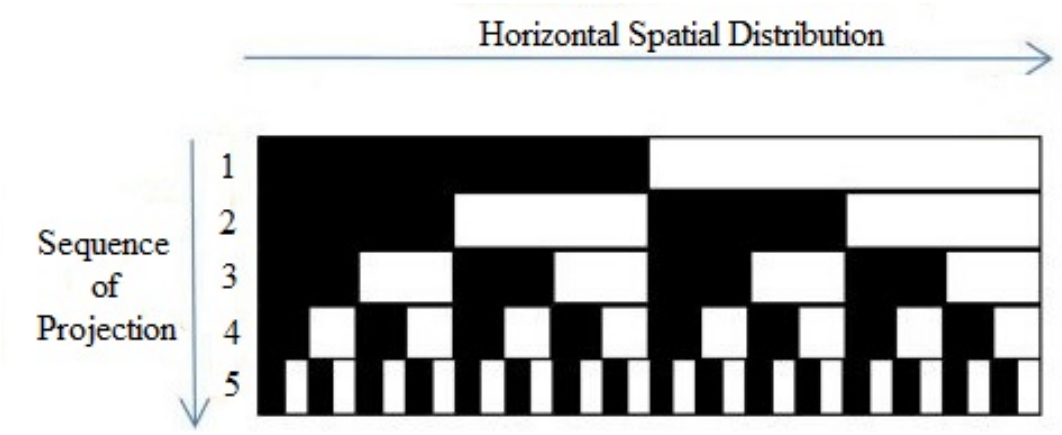


FIGURE 1.7: Code assignment for each pixel along the horizontal direction (Geng [2011])

Correspondence matching problem is thus solved. But, calibration still remains a problem. In the next section, a few existing methods of calibration are discussed.

1.5 Calibration

As discussed in section 1.3, for accurate 3D reconstruction, the intrinsic calibration matrices ($\mathbf{K}_1$ and $\mathbf{K}_2$) and extrinsic parameters ($\mathbf{R}$ and $\mathbf{t}$) of the stereo system should be estimated and this is known as calibration.

For camera calibration, any rigid body of known dimensions can be used. In figure 1.8, calibration using a rig having calibrated patterns is shown. The world origin is attached to the rig at $\mathbf{O}_w$. Each square is uniformly spaced and their dimensions are known. In the captured image, the corners of each square are detected and their pixel coordinates ($\mathbf{x}$) are estimated. The world coordinates ($\mathbf{X}$) are already known as the rig is calibrated. Using these correspondences ($\mathbf{X} \rightarrow \mathbf{x}$) the projection matrix $\mathbf{P}$, as in equation 1.8, can be estimated. This method can be used for stereo camera pair calibration as well. The method has a problem that it requires quite a bulky and properly calibrated rig which needs to be specifically designed.

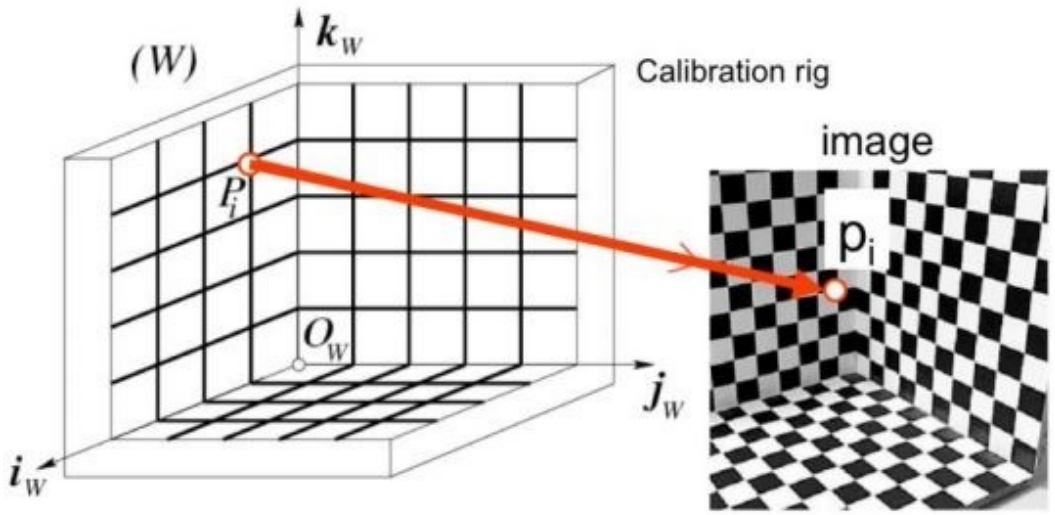


FIGURE 1.8: Camera calibration using a calibration rig.[1]

Zhang [2000] describes a method of camera calibration which is predominantly used in practice today. The proposed idea is to use a planar checkerboard pattern of known dimensions (figure 1.9) and capture its multiple images (atleast two) as it is moved so as to fill a volume (figure 1.10). Again the corners are detected and a correspondence between the corners in the image and those on the board are estimated. These correspondences, in turn, are used to estimate the calibration parameters.

There are several toolboxes available which work on this method. Figure 1.11 shows the result of camera calibration using the MATLAB Camera Calibrator App with images from figure 1.10 as its input. The relative position of the camera with respect to the position of the board is shown. It should be noted how the board has been moved to fill a volume.

This method has the distinct advantage that a bulky and specifically designed rig is not required. The pattern can be easily printed and pasted on a planar surface such as a hardcover book or a cardboard. Again, this method can be easily extended for stereo camera pair calibration.

[1]Image taken from course notes of Hata and Savarese

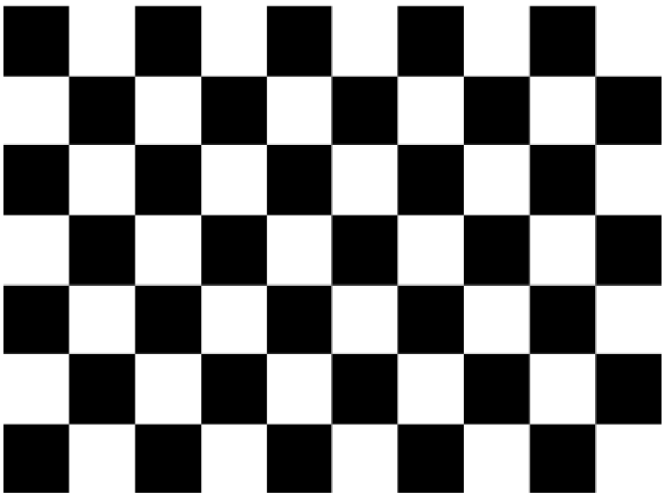


FIGURE 1.9: Checkerboard Pattern

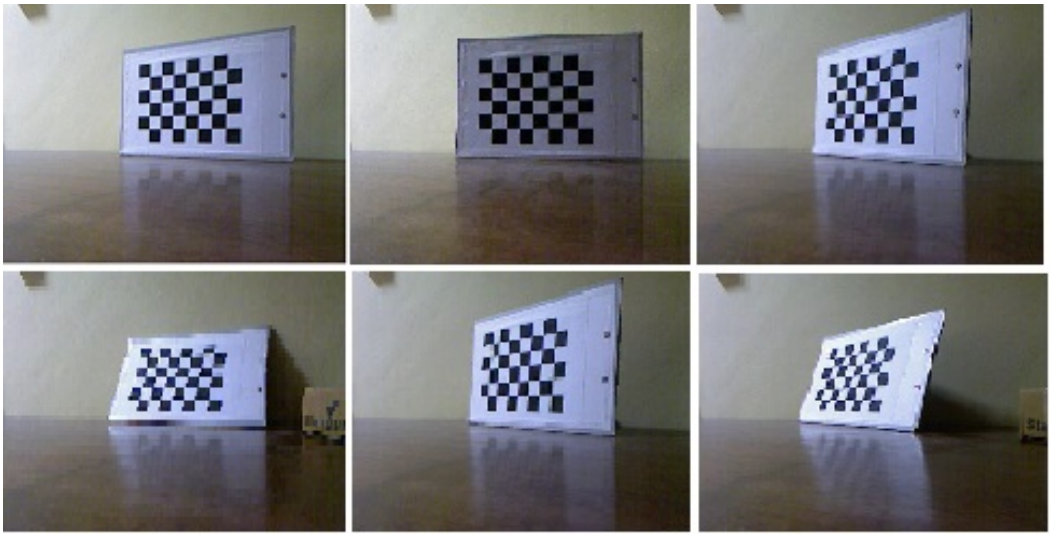


FIGURE 1.10: Camera Calibration using Zhang's Method

The methods discussed above can be employed for stereo camera pair calibration but not for SLS calibration. SLS calibration is complicated due to the presence of a projector. Though projectors can be modelled by the pinhole model, their image plane is not observable and hence SLS calibration is relatively more complicated than a stereo camera pair calibration.

A popularly used method is an extension of Zhang's method. A planar checkerboard pattern is used and a similar pattern is projected on to the board from the projector

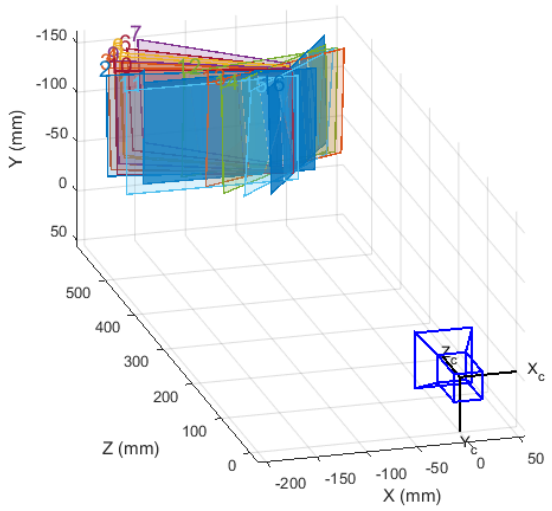


FIGURE 1.11: Camera Calibration example using the MATLAB Camera Calibrator App

(figure 1.12). Again the board is rotated so as to fill a volume. The images of the board are captured with the pattern being projected on the board. The camera is easily calibrated with the printed pattern. For the projector and system calibration, for each image:

- using the projected pattern, a homography (**H**) between the camera and the projector is estimated.
- using **H**, it is estimated where the printed pattern should be observed on the projector plane.

Once several such images as observed on the projector plane are estimated, Zhang's method can be used for the projector calibration. This method is widely used in practice today. The nature of patterns may change from implementation to implementation but for most parts the algorithm remains the same as described.

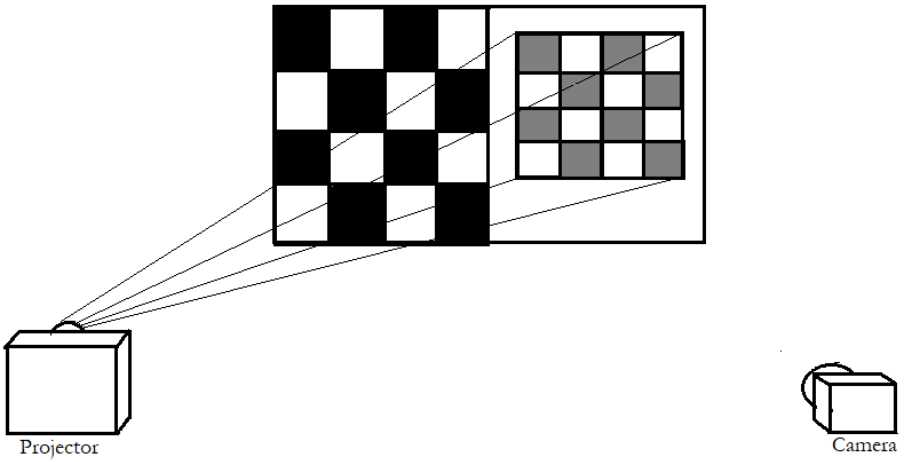


FIGURE 1.12: Structured Light System Calibration

1.6 Objectives

Figure 1.13 shows the 3D reconstruction pipeline used in most structured light systems today. Calibration is the crucial first step in this pipeline. The problem is that if the relative orientation of the camera and the projector changes, the system needs to be calibrated all over again. If a fixed structure is to be scanned the system needs to be moved a lot and so repeated calibration is required. SLS calibration as discussed previously is a tedious process and requires manual effort. So calibrating the system repetitively proves to be a cumbersome task.

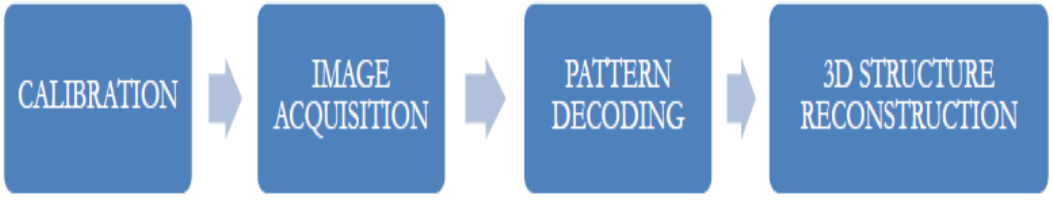


FIGURE 1.13: The conventional 3D reconstruction pipeline for structured light system

The objective of this project is to develop a new reconstruction pipeline (figure 1.14) which requires minimum manual effort. A method is proposed where the system

can be calibrated during scanning itself. A RGBD camera (e.g., Microsoft Kinect or ASUS Xtion), along with a high resolution camera and projector setup is used to achieve this goal. RGBD cameras capture depth as well as color images of the scene. The depth data along with the information obtained in color images from the projected patterns is used for system calibration as well as 3D reconstruction.

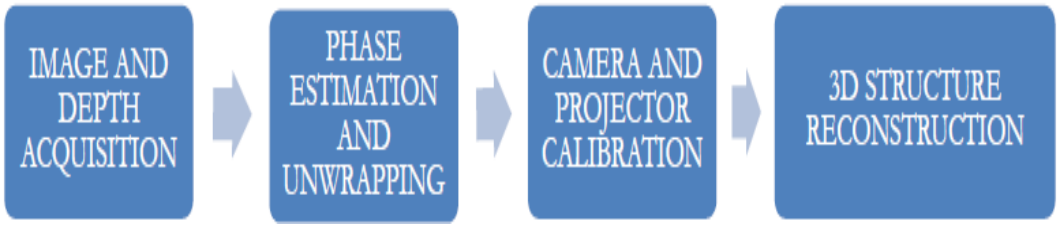


FIGURE 1.14: The proposed 3D reconstruction pipeline

Apart from developing a new pipeline for 3D reconstruction, issues of accuracy, noise, radiometric non-linearity, and computation time have been experimentally studied and discussed in this report. In chapter 2, each step in the pipeline and its associated mathematics have been explained. In chapter 3, the qualitative and quantitative evaluation of the 3D scans is presented. In chapter 4, this report is concluded with inferences from the results, the advantages and disadvantages of this method, and some suggested future scope of this project.

Chapter 2

Methodology

As stated earlier, the primary objective of this project is to develop a structured light 3D scanner that requires minimum calibration effort and produces high quality scans. This chapter discusses the method employed to achieve this goal.

RGBD or depth cameras can be used as 3D scanners. They are easily available in the market, are calibrated during production, and need not be calibrated before each scan. Though the scans from these cameras are accurate, they are not of very high quality. Many of the finer details are averaged out as seen in figure 2.1. But the depth data can be fused with the concept of structured light to calibrate the structured light system without any manual effort.

FIGURE 2.1: Image on the right shows 3D scan of the idol on the left using Kinect

The algorithm developed for implementing the pipeline (figure 1.14) is described in gist below:

- The depth map is captured using the RGBD camera.
- RGB images with the projected patterns are captured using the RGBD camera and the high resolution camera.
- The patterns are decoded to estimate the phase maps for both the cameras.
- Using the 3D coordinates and the phase maps from the RGBD camera, the projector is calibrated with respect to the RGBD camera and the projection matrix $\mathbf{P}_{pk}$ is estimated.
- An improved depth map is obtained with the RGBD camera phase map and the projection matrix $\mathbf{P}_{pk}$.
- Correspondences between the two cameras are found using their phase maps.
- With these correspondences and the new depth map, the high resolution camera is calibrated with respect to the RGBD camera and the projection matrix $\mathbf{P}_{ck}$ is estimated.
- $\mathbf{P}_{ck}$ is decomposed to obtain the extrinsic parameters ($\mathbf{R}$ and $\mathbf{t}$) as well as the intrinsic calibration matrix ($\mathbf{K}_c$).
- With $\mathbf{R}$ and $\mathbf{t}$, the world reference frame is shifted from the RGBD camera to the high resolution camera.
- With 3D coordinates in camera reference frame and the camera phase maps, the projector is calibrated with respect to the high resolution camera and the projection matrix $\mathbf{P}_{pc}$ is estimated.
- With $\mathbf{P}_{pc}$ and the camera phase map, the high resolution depth map is estimated.

2.1 Data Acquisition

The first step in the pipeline is capturing the depth data and the RGB images with the projected patterns.

2.1.1 Depth Map

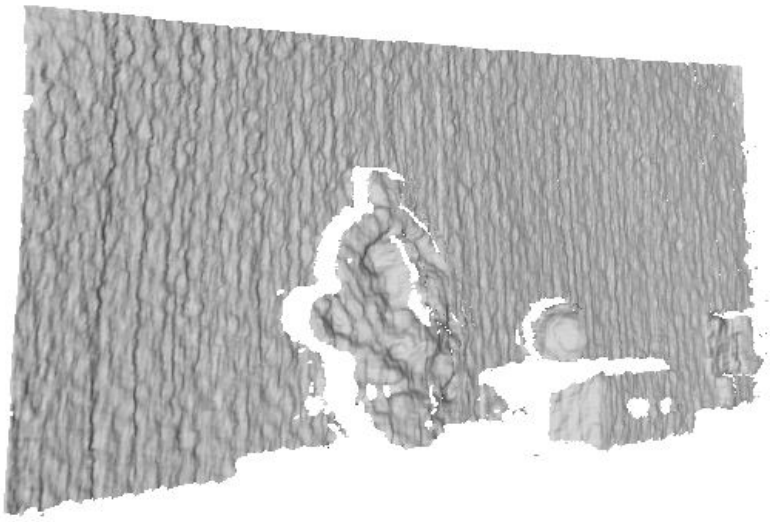


FIGURE 2.2: Depth Map acquired using Kinect

Figure 2.2 shows a captured depth map. The depth map and the RGB images from the RGBD camera are registered with each other, that is, they have pixel to pixel correspondence. Thus, it is known which 3D point forms the image at which pixel in the RGB image.

2.1.2 Pattern Selection

An important factor to be considered in structured light scanning is the type of pattern to be used for encoding the view space. As discussed in section 1.4, there are quite a few different coding strategies in use. Salvi et al. [2004] have summarized many of these strategies along with their advantages and disadvantages.

In this implementation, sinusoidal phase shifted structured light patterns (figure 2.3) have been used. The advantages are subpixel resolution, reduced sensitivity to diffusion caused by any focusing issues in the projector, and fewer number of patterns are required (as few as three). The disadvantage is that with fewer number of patterns, ripples may appear in the scans due to radiometric non-linearity.

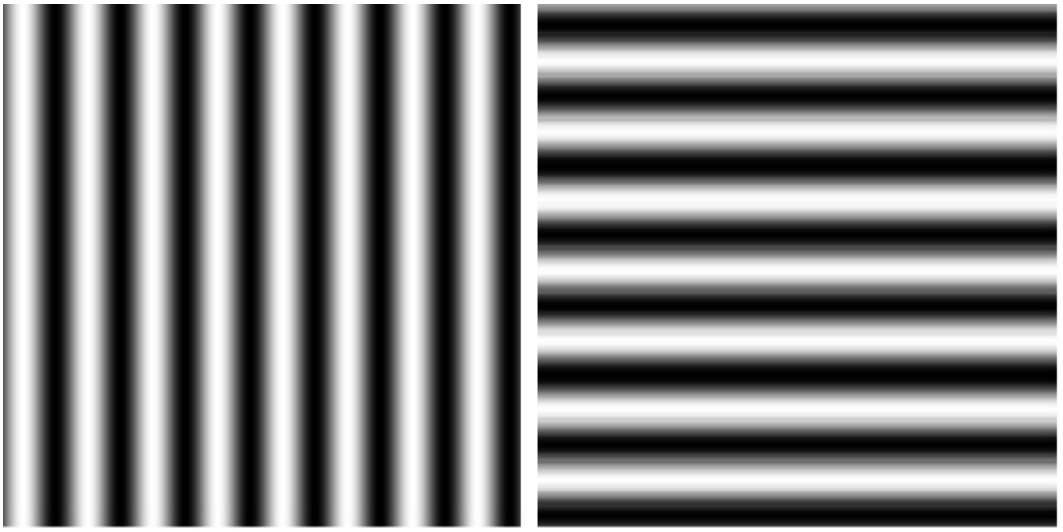


FIGURE 2.3: Example of sinusoidal phase shifted patterns for horizontal (left) and vertical (right) encoding

The patterns for horizontal (along the x-axis) encoding are generated using the below equation:

$$I_f^k(\mathbf{x}) = a + b\cos(\frac{2\pi f x}{width} + \frac{2\pi k}{N}) \tag{2.1}$$

where $I_f^k(\mathbf{x})$ is the intensity at the pixel $\mathbf{x}$ for the k^{th} phase shift and frequency f. The constant a is chosen so as to keep the grayscale level above zero and b is the amplitude. Both are typically chosen to be 0.5. x is the x coordinate of the pixel $\mathbf{x}$, and N is the total number of phase shifts. For vertical (along the y-axis) encoding, x is replaced by y and $width$ is replaced by $height$. The resolution of projected patterns being $width \times height$. Figure 2.4 shows two images captured by projecting the patterns for horizontal and vertical encoding respectively.

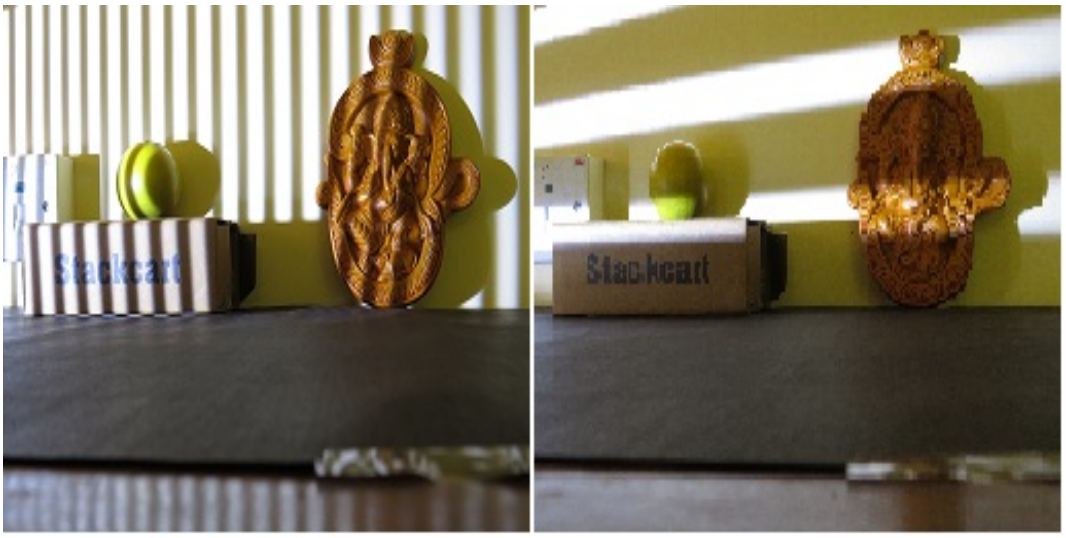


FIGURE 2.4: Example images captured with the projected patterns

In this implementation, six frequencies from 1 to 32 cycles have been used. A higher resolution is achieved with the higher frequencies. The lower frequencies are needed for phase unwrapping (section 2.2.2).

For each frequency, eight phase shifts have been used. In all, 96 images (48 for horizontal encoding and 48 for vertical) need to be captured using the RGBD camera and the high resolution camera each. The effects of changing the number of shifts have been studied and discussed in section 3.2.2.2.

2.2 Pattern Decoding

The acquired images are first converted from RGB to grayscale. Next, for each frequency, the images for the eight phase shifts are processed to estimate a phase map. The six phase maps corresponding to each of the frequencies are then processed to obtain unique codes for each of the pixels.

2.2.1 Phase Estimation

For each frequency f, the eight grayscale images are processed to obtain the phase value ($\theta_f(\mathbf{x})$) at pixel $\mathbf{x}$ as:

$$\theta_f(\mathbf{x}) = \tan^{-1}(-\frac{\sum_{k=0}^{N-1} I_f^k(\mathbf{x}) \sin(\frac{2\pi k}{N})}{\sum_{k=0}^{N-1} I_f^k(\mathbf{x}) \cos(\frac{2\pi k}{N})}) \tag{2.2}$$

The estimated phase maps for two of the frequencies are shown in figures 2.5 and 2.6. It can be observed that the pixels do not yet have unique codes. The phase is wrapped to the range $(-\pi, \pi)$. For unique code assignment, the phase should be unwrapped.

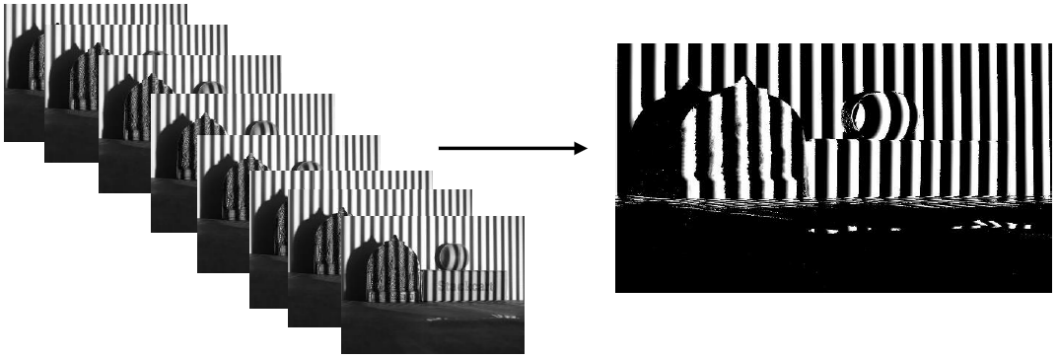


FIGURE 2.5: Estimated phase for horizontal coding for f= 32 cycles.

2.2.2 Phase Unwrapping

The estimated phase in equation 2.2, can also be expressed as:

$$\theta_f(\mathbf{x}_c) = \Phi(\mathbf{x}_p) - 2\pi\lfloor\Phi(\mathbf{x}_p)/2\pi\rfloor \tag{2.3}$$

where $\mathbf{x}_p$ is the projector pixel corresponding to the camera pixel $\mathbf{x}_c$ and $\Phi(\mathbf{x}_p)$ is the phase at pixel $\mathbf{x}_p$. $\theta_f(\mathbf{x}_c)$ should be unwrapped so that the unwrapped phase at $\mathbf{x}_c$, $\Theta(\mathbf{x}_c) = \Phi(\mathbf{x}_p)$.

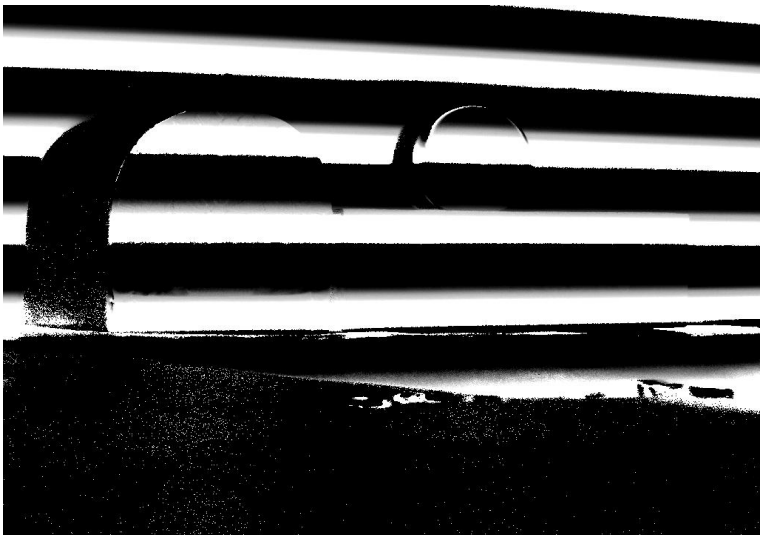


FIGURE 2.6: Estimated phase for vertical coding for f = 8 cycles

There are several unwrapping algorithms that can be used for phase unwrapping. Salvi et al. [2004] describe a method which fuses sinusoidal patterns with gray code patterns. Each projector pixel is assigned a gray code that has an equivalent code of $k(\mathbf{x}_p) = \lfloor \Phi_f(\mathbf{x}_p)/2\pi \rfloor$. The unwrapped phase value at each camera pixel is then $\Theta(\mathbf{x}_c) = \theta_f(\mathbf{x}_c) + 2\pi k(\mathbf{x}_c)$. The total number of patterns are significantly reduced with this method but a drawback is that accurate alignment of the phase shifted patterns and the gray coded patterns must be ensured. Another problem is that gray coded patterns are prone to blur. Also, since a single frequency is used for phase shifted patterns, errors in phase estimation cannot be addressed. Hence, this method of phase unwrapping has not been used.

In this project, six frequencies (1 to 32 cycles) in multiples of 2 have been used to unwrap the phase. The base frequency is only one cycle, so it is unwrapped by adding 2π to the negative phase values. Since the frequencies are multiples of two, the unwrapped phase value at a pixel $\mathbf{x}_c$ for a given frequency should be exactly twice the unwrapped phase value at the same pixel for the immediately lower frequency. Mathematically this can be expressed as:

$$\Theta_f(\mathbf{x}_c) = \theta_f(\mathbf{x}_c) + 2\pi \lfloor \frac{2\Theta_{f-1}(\mathbf{x}_c) - \theta_f(\mathbf{x}_c)}{2\pi} \rceil \tag{2.4}$$

$\theta_f(\mathbf{x}_c)$ is the wrapped phase and $\Theta_f(\mathbf{x}_c)$ is the unwrapped phase at pixel $\mathbf{x}_c$ for frequency f, and $\lfloor\rceil$ denotes rounding off to nearest integer. Phase is unwrapped iteratively for the frequencies of 2 to 32 cycles on the basis of the base frequency of 1 cycle. Figures 2.7 and 2.8 show the unwrapped phase maps for horizontal and vertical coding respectively.

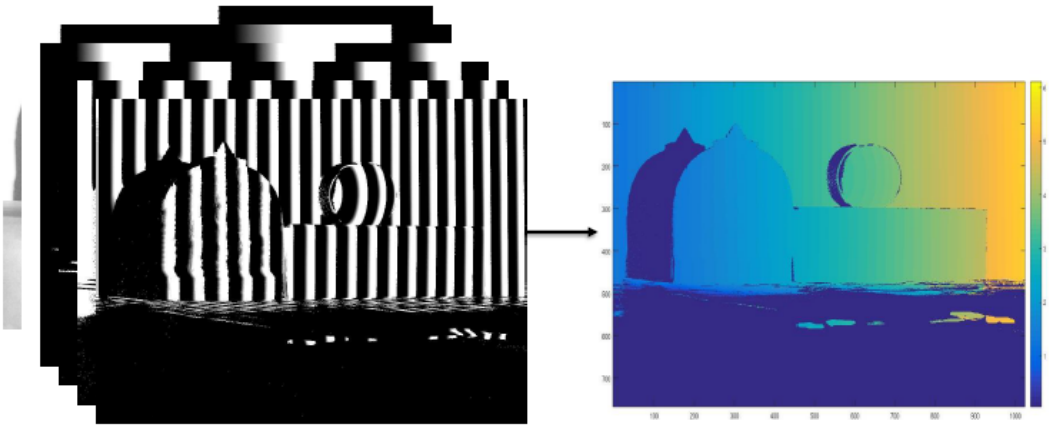


FIGURE 2.7: Unwrapped phase for horizontal coding

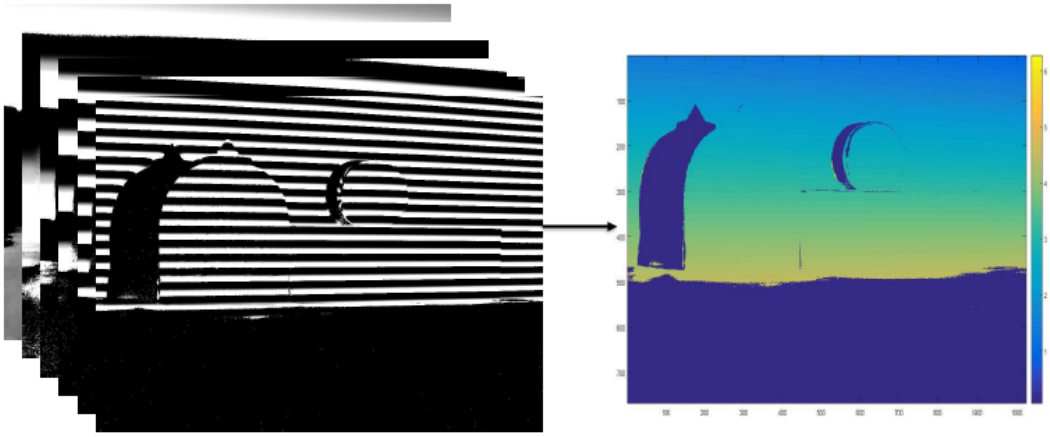


FIGURE 2.8: Unwrapped phase for vertical coding

Two sets of phase maps are thus estimated. One set each for the RGBD camera $(\Theta_{k,x}, \Theta_{k,y})$ and the high resolution camera $(\Theta_{c,x}, \Theta_{c,y})$. Before finding any correspondences, the phase maps are filtered using low pass Gaussian filter to remove any random noise in the estimated phase. The estimated pair of phase values (Θ_x, Θ_y) at a pixel directly gives the corresponding projector pixel coordinates. Since the

depth map and the RGB images from the RGBD camera have pixel to pixel correspondences, the phase values at each 3D point that forms an image is known. Thus, as shown in figure 2.9, the four phase maps and the depth maps establish all the correspondences required for calibrating the devices and the system.

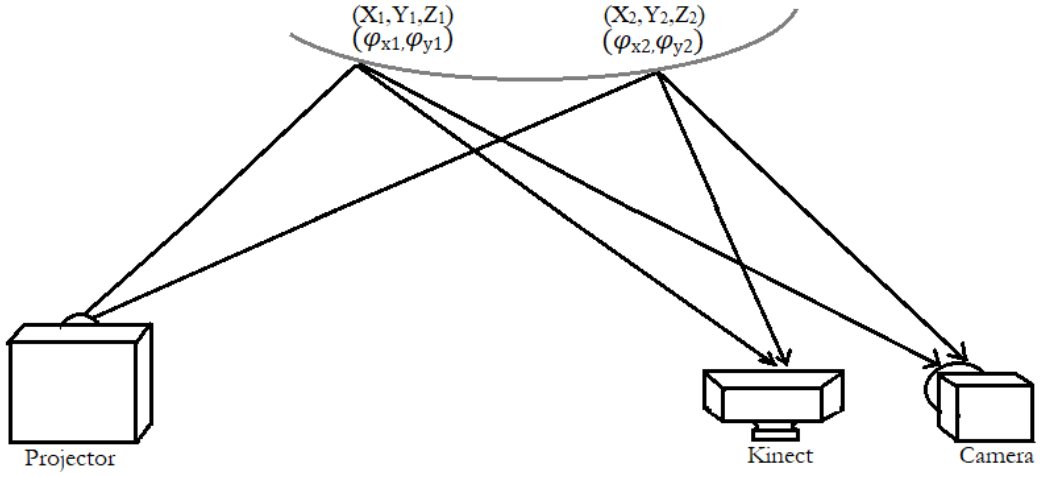


FIGURE 2.9: Correspondence matching between the three devices

2.3 Calibration

With the four phase maps and the depth map, it is possible to calibrate the projector, the camera and the structured light system. The aim here is to estimate the projection matrix **P** as in equation 1.8. This section describes the method of calibration used in this implementation.

2.3.1 Direct Linear Transformation (DLT)

The projective relationship between a 3D point and its image as given by equation 1.8, can be rewritten as:

$$\mathbf{x} \propto \begin{bmatrix} \mathbf{P}_{r1}^T \\ \mathbf{P}_{r3}^T \\ \mathbf{P}_{r3}^T \end{bmatrix} \mathbf{X} \tag{2.5}$$

where $\mathbf{x}$ is the homogeneous pixel coordinate $(x, y, 1)$, $\mathbf{X}$ is the homogeneous 3D coordinate $(X, Y, Z, 1)$, and $\mathbf{P}_{ri}^T$ is the i^{th} row of the 3×4 projection matrix $\mathbf{P}$. Since the relationship in equation 2.5 is proportional up to a scale, it cannot be solved for directly. Because of the proportionality, their cross product should be zero:

$$\mathbf{x} \times \mathbf{PX} = \mathbf{0} \tag{2.6}$$

which can further be written as:

$$\mathbf{P}_{r1}^T\mathbf{X} - x\mathbf{P}_{r3}^T\mathbf{X} = 0 \tag{2.7}$$

$$\mathbf{P}_{r2}^T\mathbf{X} - y\mathbf{P}_{r3}^T\mathbf{X} = 0 \tag{2.8}$$

Equations 2.7 and 2.8 can be combined as:

$$\begin{bmatrix} \mathbf{X}^T & \mathbf{0}_{1\times4} & -x\mathbf{X}^T \\ \mathbf{0}_{1\times4} & \mathbf{X}^T & -y\mathbf{X}^T \end{bmatrix} \begin{bmatrix} \mathbf{P}_{r1} \\ \mathbf{P}_{r2} \\ \mathbf{P}_{r3} \end{bmatrix} = \mathbf{0}_{12\times1} \tag{2.9}$$

The matrix $\mathbf{P}$ has 11 degrees of freedom. Since there are two equations for each $\mathbf{X} \rightarrow \mathbf{x}$ mapping, at least five and a half such correspondences are required for estimating $\mathbf{P}$. $\mathbf{P}$ is the right null vector of the matrix on the left and can be estimated using SVD.

DLT is a linear method of estimating $\mathbf{P}$. To ensure the stability of the method, the 3D and the pixel coordinates are first normalized (Appendix B). Even then, the estimated $\mathbf{P}$ is rarely accurate enough. A better approach for estimating $\mathbf{P}$ should be used.

2.3.2 Reprojection Error Minimization

Since the estimate of $\mathbf{P}$ using DLT is not accurate enough, a method based on minimizing the reprojection error is used.

Reprojection Error

Let $\mathbf{X} \rightarrow \mathbf{x}$ be a known correspondence. If $\mathbf{X}$ is reprojected on to the image plane using the estimated projection matrix $\mathbf{P}$, such that the image of $\mathbf{X}$ is now formed at $\mathbf{x}'$, the euclidean distance between $\mathbf{x}$ and $\mathbf{x}'$ is known as reprojection error (d in figure 2.10).

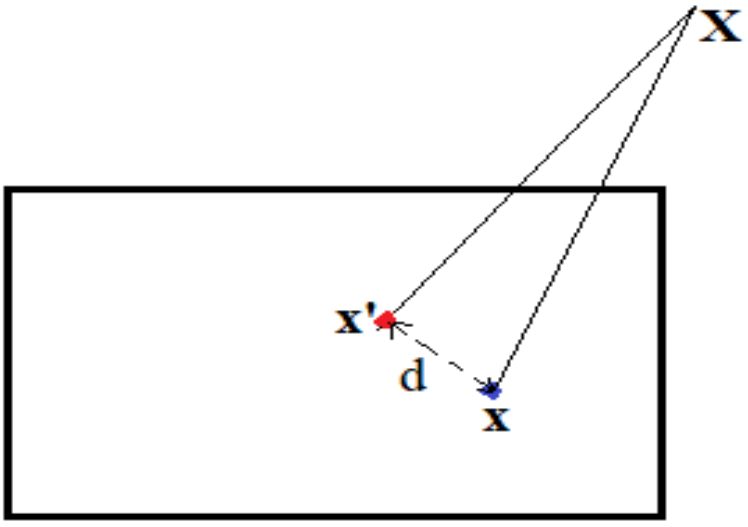


FIGURE 2.10: Reprojection Error

From equations 2.7 and 2.8, reprojection error can be expressed as:

$$d = \sqrt{(x - \frac{\mathbf{P}_{r1}^T\mathbf{X}}{\mathbf{P}_{r3}^T\mathbf{X}})^2 + (y - \frac{\mathbf{P}_{r2}^T\mathbf{X}}{\mathbf{P}_{r3}^T\mathbf{X}})^2} \tag{2.10}$$

The sum of reprojection errors for all the point correspondences is used as the cost function to be minimized for estimating $\mathbf{P}$. The cost function is non-linear in the

parameters to be estimated. The estimate from DLT is used as the initial estimate of $\mathbf{P}$ for the cost function minimization.

Even with reprojection error minimization, the estimated projection matrix is not accurate enough because of outliers in the acquired data. To better estimate $\mathbf{P}$ in the presence of outliers, random sample consensus (RANSAC) is used.

2.3.3 Random Sample Consensus (RANSAC)

RANSAC is an iterative method that estimates the parameters of a mathematical model from a set of data that has outliers. It was first published by Fischler and Bolles [1981] at SRI International. Since then it has been used in various applications.

In this implementation, the inputs to the RANSAC algorithm are:

- Data : The $\mathbf{X} \rightarrow \mathbf{x}$ correspondences for the device to be calibrated.
- Number of data points (n) required to fit a model : At least five and a half points are required. But, to accurately estimate the projection matrix, 100 points are sampled in each iteration.
- Fitting function : With the sampled data, $\mathbf{P}$ is first estimated using DLT. This estimate is then used to initialize the reprojection error minimization. Thus, DLT folowed by the minimization is used as the fitting function.
- Distance function : Reprojection error is used as the criteria on the basis of which outliers and inliers are classified.
- Threshold value for classification between outliers and inliers : With the normalized data (section 2.3.1), the correspondences with reprojection error above 0.05 are classified as outliers.

- Probability of finding an inlier in the data (p) : With several experiments and the analysis of the data, this probability was set to be 0.85.

The outputs from RANSAC are:

- Model having the largest number of inliers : The estimated projection matrix **P**.
- The inliers corresponding to the estimated model.

RANSAC randomly samples the data for n points and fits a model. With the estimated model, it uses the distance function and the threshold to find the number of inliers. It keeps a record of the model with maximum inliers. This is done iteratively until the termination criteria is met or the maximum number of iterations are reached.

An accurate estimate of the projection matrix is achieved using the calibration procedure described above. Next, it is necessary to decompose the projection matrix to estimate the intrinsic and extrinsic calibration matrices.

2.3.4 Decomposing the Projection Matrix

As shown in equation 1.8, the projection matrix is composed of the intrinsic calibration matrix and the extrinsic calibration matrix. Intrinsic calibration matrix contains information such as focal length and the coordinates of the principal point, which are necessary for stereo triangulation. The extrinsic calibration matrix contains the rotation matrix and the translation vector which are required for changing the world reference frame for calibration of other devices.

2.3.4.1 Estimating the translation vector (t)

A better parameter to estimate instead of the translation vector ($\mathbf{t}$) is the camera center ($\mathbf{O}$). The camera center is related to the translation vector as:

$$\mathbf{RO} + \mathbf{t} = \mathbf{0} \tag{2.11}$$

$$\implies \mathbf{O} = -\mathbf{R}^{-1}\mathbf{t} \tag{2.12}$$

Also, the camera center is related to the projection matrix as:

$$\mathbf{RO} + \mathbf{t} = \mathbf{0} \tag{2.13}$$

$$\implies \mathbf{K}(\mathbf{RO} + \mathbf{t}) = \mathbf{0} \tag{2.14}$$

$$\implies \mathbf{K}\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix}\mathbf{O}_h = \mathbf{0} \tag{2.15}$$

$$\implies \mathbf{P}\mathbf{O}_h = \mathbf{0} \tag{2.16}$$

This implies that the homogeneous camera center ($\mathbf{O}_h$) is the right null vector of the projection matrix ($\mathbf{P}$) and can be estimated with SVD of $\mathbf{P}$. Once the camera center is estimated, the translation vector can be estimated as shown above.

2.3.4.2 Estimating the intrinsic calibration matrix (K) and the rotation matrix (R)

For a finite camera, we have:

$$\mathbf{P} = \mathbf{K}\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix} = \mathbf{K}\begin{bmatrix}\mathbf{R} & -\mathbf{RO}\end{bmatrix} = \begin{bmatrix}\mathbf{M} & -\mathbf{MO}\end{bmatrix} \tag{2.17}$$

This implies that the product of the matrices $\mathbf{K}$ and $\mathbf{R}$ is proportional to the first 3×3 elements of the projection matrix $\mathbf{P}$ which is denoted by $\mathbf{M}$ in the above equation. With $\det(\mathbf{M}) > 0$, $\mathbf{K}$ and $\mathbf{R}$ can be calculated by the RQ factorization (Appendix A) of $\mathbf{M}$. The RQ factorization of a matrix $\mathbf{A}$ gives an upper triangular matrix $\mathbf{K}$

("**R**") and an orthogonal matrix **R** ("**Q**") such that $\mathbf{KR} = \mathbf{A}$. If **K** is chosen to have positive diagonal elements, the factorization is unique. If the calibration matrix **K** is scaled such that $K_{33} = 1$, it may be interpreted as:

$$\mathbf{K} = \begin{bmatrix} \alpha & \gamma & c_x \\ 0 & \beta & c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{2.18}$$

Thus the projection matrix $\mathbf{P}_{ck}$ is decomposed to obtain rotation matrix (**R**) and the translation vector (**t**). With this **R** and **t**, the world reference frame is shifted from the RGBD camera to the high resolution camera:

$$\mathbf{X}_c = \mathbf{R}\mathbf{X}_k + \mathbf{t} \tag{2.19}$$

where $\mathbf{X}_c$ and $\mathbf{X}_k$ are non-homogeneous world coordinates with respect to the high resolution camera and the RGBD camera coordinate frames respectively, and **R** and **t** are the rotation and translation of the high resolution camera with respect to the RGBD camera. With the 3D coordinates $\mathbf{X}_c$, the projector is calibrated with respect to the camera and the projection matrix ($\mathbf{P}_{pc}$) is estimated. With $\mathbf{P}_{pc}$ and the camera phase map for horizontal encoding, the 3D structure can be estimated as described next.

2.4 Estimating the 3D structure

With the projection matrix of the projector with respect to the camera ($\mathbf{P}_{pc}$) and the horizontal phase map (for encoding along x axis) in the camera view, the depth map is estimated using ray-plane triangulation (figure 2.11). Each camera pixel gives a ray direction and the phase value at each pixel gives the identity of the plane with which that ray intersects.

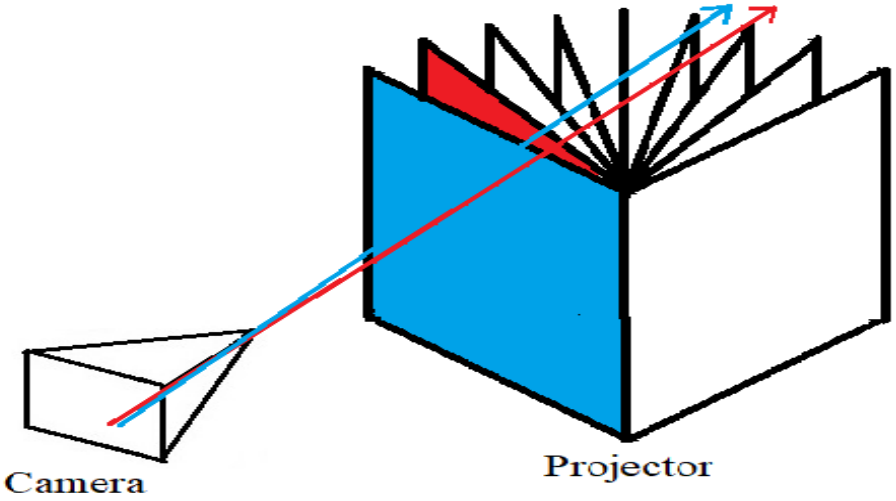


FIGURE 2.11: Ray Plane Triangulation

For ray-plane triangulation, the whole projection matrix is not required. As can be seen in figure 2.11, the projector axis is enough for triangulation i.e, the projector center is not required. Therefore, for ray-plane triangulation, we have:

$$\begin{bmatrix} x_p \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{P}_{r1}^T \\ \mathbf{P}_{r3}^T \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{2.20}$$

Given the camera focal length (f_c), X and Y are related to the camera pixel coordinates (x_c, y_c) as:

$$X = \frac{x_c Z}{f_c} \tag{2.21}$$

$$Y = \frac{y_c Z}{f_c} \tag{2.22}$$

Hence, with f_c, x_c, y_c, and x_p known, the depth can be estimated from equation 2.20 as:

$$\begin{bmatrix} x_p \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{P}_{r1}^T \\ \mathbf{P}_{r3}^T \end{bmatrix} \begin{bmatrix} \frac{x_c Z}{f_c} \\ \frac{y_c Z}{f_c} \\ Z \\ 1 \end{bmatrix} \tag{2.23}$$

where x_p is the estimated phase value at the camera pixel (x_c, y_c) and indicates the plane with which the ray through (x_c, y_c) intersects.

With the estimated depth, the corresponding X and Y coordinates are calculated using equations 2.21 and 2.22. Delaunay triangulation is used to estimate the mesh from the 3D points. A detailed discussion of the 3D scans thus obtained, along with discussions on accuracy, effect of baseline and number of patterns, and computation time are presented in the next chapter.

Chapter 3

Results and Discussion

There were two main phases in the development of this project:

- Improvement of the 3D scans from the RGBD camera using structured light. The relevant results have been discussed in section 3.1.
- RGBD cameras, in general, are limited in resolution and cannot be zoomed in to focus on the object. Hence, the next step was to include a high resolution camera to increase the resolution and capture more details. The scans from the high resolution camera are discussed in section 3.2.

The effect of baseline and number of patterns on the scans have been discussed as well. Figure 3.1 shows the experimental setup of the 3D scanner. The hardware used in setting up this 3D scanner are:

- Microsoft Kinect for XBOX 360
- Canon PowerShot SX540HS Camera
- Egate EG i9 Projector

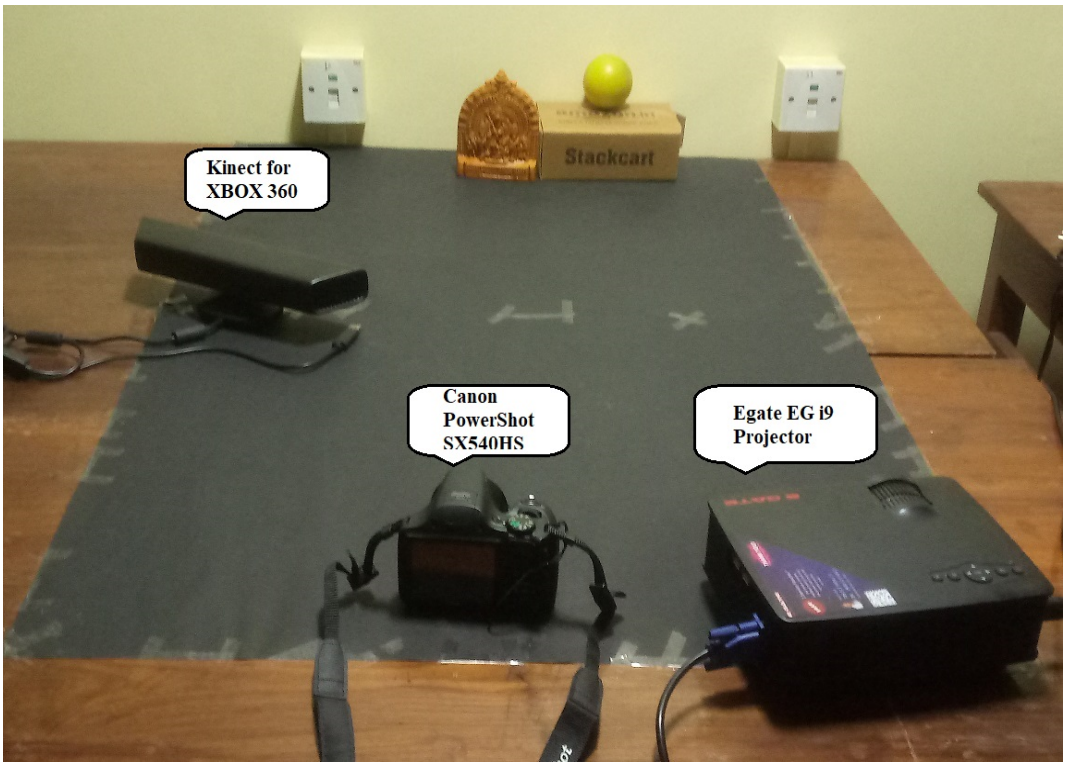


FIGURE 3.1: Setup of the 3D Scanner

The resolution of Kinect was 640×480, the camera was set at a resolution of 2048×1536, and the projector was set at a resolution of 1366×768. To ensure the accuracy of the scans, Kinect was calibrated using Zhang's method (Zhang [2000]) with MATLAB's camera calibrator application. The software used in this project are:

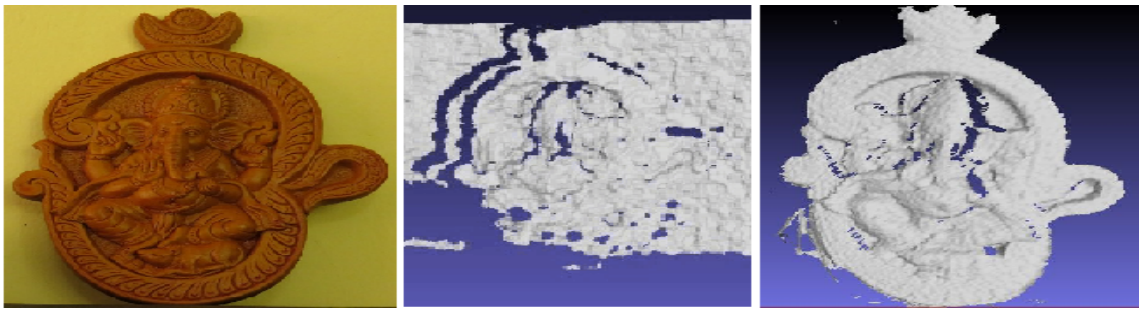
- MATLAB 2017a for algorithm implementation
- NiViewer to capture data using Kinect
- Meshlab (Cignoni et al. [2008]) for viewing the 3D scans

3.1 Improvement of Kinect Scans

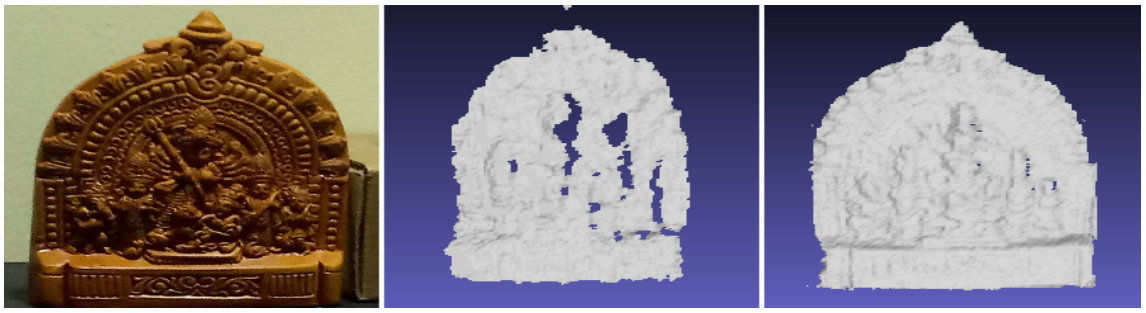
The phase maps estimated with the images from the RGB camera of Kinect give the corresponding projector pixel location. With the depth map and the projector pixel correspondences, the projector is calibrated with respect to Kinect to obtain the projection matrix $\mathbf{P}_{pk}$ as described in section 2.3. With $\mathbf{P}_{pk}$ and the phase map for horizontal encoding, a refined 3D structure is estimated using ray-plane triangulation (section 2.4).
Figure 3.1 shows the original and the refined scans along with the scanned objects. It can be immediately seen that the refined scans have much more structure and detail than the original scans.

3.2 High Resolution Camera Scans

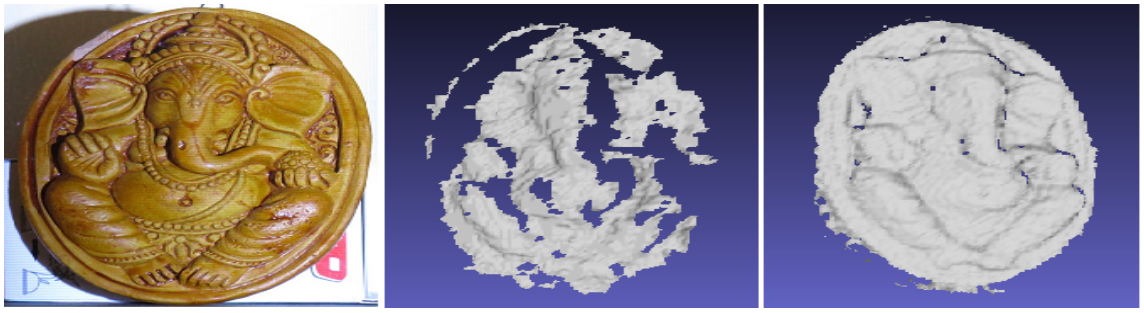
Even though the refined scans, as shown in the previous section, are much better than the original scans, they are not as detailed as the objects themselves. This is because Kinect is limited in resolution and it cannot be zoomed in to focus on the object. As stated earlier, a high resolution camera with image resolution of 2048×1536 is used along with Kinect and the projector to get high quality scans. Each 3D point as observed from Kinect has a pair of phase values attached. The camera pixel where these pair of phase values are observed gives the location of the image on the camera image plane of the corresponding 3D point. With these correspondences, the camera is calibrated with respect to Kinect to estimate the projection matrix $\mathbf{P}_{ck}$ (section 2.3). This $\mathbf{P}_{ck}$ is then decomposed (section 2.3.4) to obtain the rotation matrix ($\mathbf{R}$) and the translation vector ($\mathbf{t}$) of the camera with respect to Kinect. Using this $\mathbf{R}$ and $\mathbf{t}$, the coordinate frame is shifted from Kinect to the camera using equation 2.19. With the 3D point coordinates in the camera reference frame, the projector is calibrated (section 2.3) with respect to the camera to estimate the projection matrix $\mathbf{P}_{pc}$. With $\mathbf{P}_{pc}$ and the camera phase map for



(a) Idol of Lord Ganesha (230 mm×170 mm)



(b) Idol of Goddess Durga (140 mm×110 mm)



(c) A smaller idol of Lord Ganesha (130 mm×102 mm)

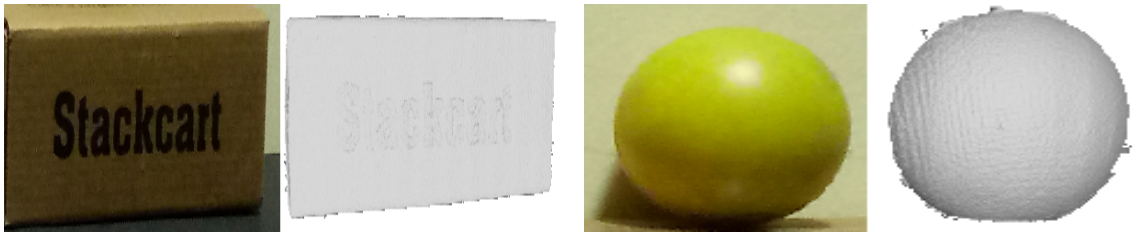
FIGURE 3.2: Results showing improvement in Kinect scans using structured light.

horizontal encoding (along the x axis), a high resolution and detailed 3D scan is then estimated using ray-plane triangulation (section 2.4). A detailed evaluation of the estimated 3D models is presented next.

3.2.1 Qualitative Evaluation

Objects with varying amount of details have been scanned. Figure 3.3 shows scans for simple objects. These simple objects seem to be accurately reconstructed. Figure 3.4 shows scans of relatively complex idols of Lord Ganesha and Goddess Durga.

These idols have significant amount of details which have been recovered accurately in the scans. The scan of a larger idol of Lord Ganesha is shown in figure 3.5 with the details zoomed in for better visibility. In all these scans, it can be observed that the fine detailing on the idols like etchings on the legs, hands, and ears, and the small patterns are faithfully reconstructed.



(a) Box (63 mm × 190 mm) (b) Ball (radius = 6cm)

FIGURE 3.3: Scans of simple objects.

(a) Idol of Lord Ganesha (130 mm × 102 mm) (b) Idol of Goddess Durga (140 mm × 110 mm)

FIGURE 3.4: Scans of more complex objects.

3.2.2 Quantitative Evaluation

To evaluate the accuracy of reconstruction, six distances as shown in figure 3.6 were physically measured. The same set of distances were measured from the estimated 3D model by locating the end points. Distances along all the three coordinate axes have been measured to study the accuracy of reconstruction. The comparison is

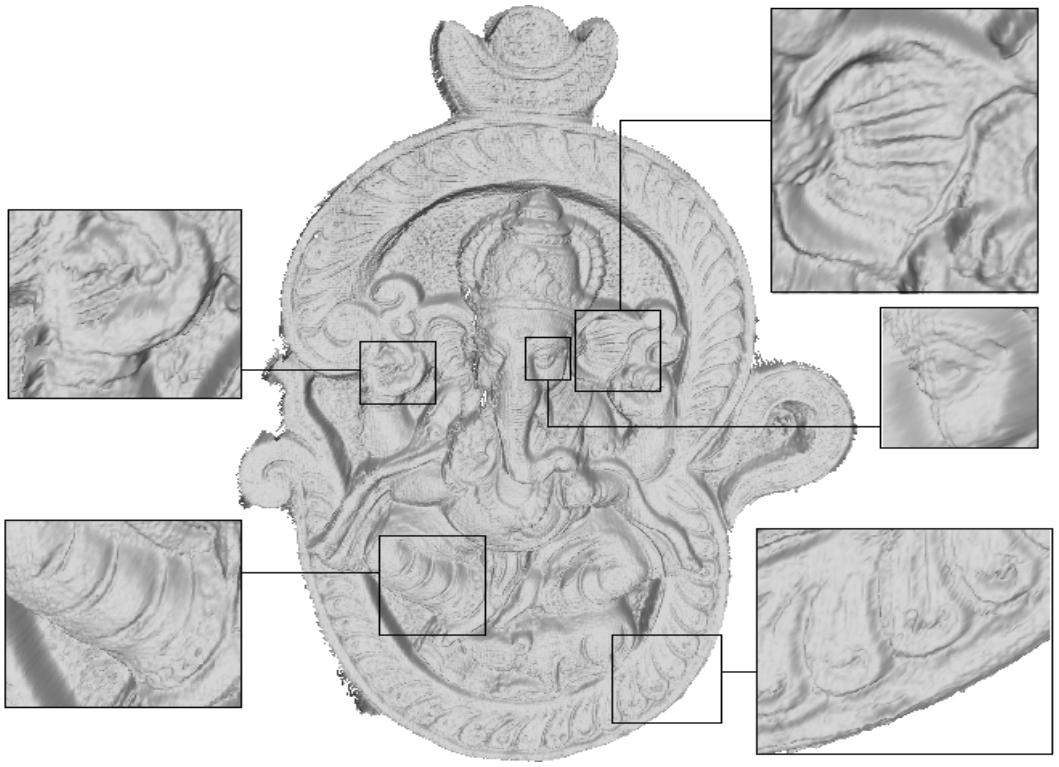
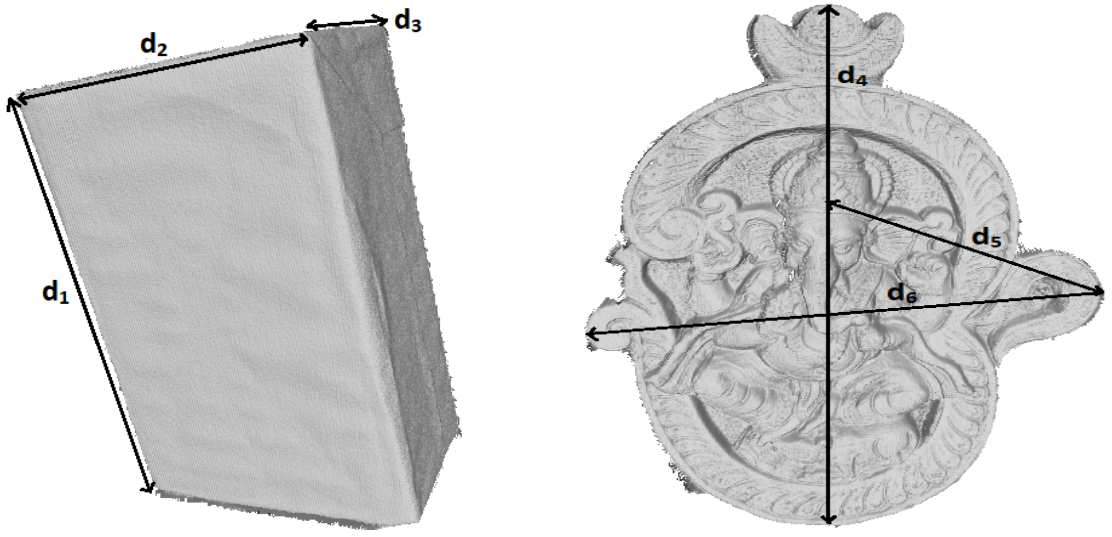


FIGURE 3.5: Scan of a terracotta idol of Lord Ganesha (230 mm × 170mm).

presented in table 3.1. The absolute differences in the estimated distances are on the order of 1 mm which establishes that these reconstructions are highly accurate.



(a) Scan of a box. (b) Scan of an idol of Lord Ganesha.

FIGURE 3.6: Scans with the measured distances marked.

Source	d_1	d_2	d_3	d_4	d_5	d_6
Physical Object	190	129	63	230	100	170
Reconstructed Model	191.24	130.11	63.39	228.55	99.51	168.84
$\|Error\|$	1.24	1.11	0.39	1.45	0.49	1.16

TABLE 3.1: Distances (in mm) measured physically and from the estimated models along with absolute errors.

3.2.2.1 The Effect of Baseline on Accuracy

The measurements of the 3D model presented in table 3.1 were initially found to be inaccurate by 4-5%. After analysing the results and the experimental setup, we concluded that the inaccuracies were due to a small baseline between the camera and the projector. While a larger baseline means that the 3D points will be well triangulated, occlusion also increases with increase in the baseline. To minimize occlusion while ensuring accurate triangulation, the 3D point estimates from Kinect are employed to correct the 3D estimates from the camera view.

Kinect is well calibrated and its focal length, image resolution, etc. do not change. Also, since the primary purpose of Kinect in the SLS setup is system calibration, it can be kept at a larger distance (baseline) with respect to the camera and the projector. This ensures that the refined 3D estimates from Kinect are accurate. Also, as stated earlier, Kinect was calibrated with Zhang's method (Zhang [2000]) using the MATLAB camera calibrator application to ensure the accuracy. To minimize occlusion, the camera is kept at a smaller baseline with respect to the projector.[1]

A 3D→3D projective transformation based method is proposed to correct the 3D estimates from the camera view. Assuming the 3D estimates from the camera view

[1]Haque et al. [2014] propose a fusion algorithm of depth and normal estimates. A similar approach with fusion of depths from two different views could be used but there is a large difference between the number of points from the Kinect view and the camera view. So, a working implementation of this method could not be developed.

are somewhat distorted with respect to the accurate estimates from Kinect view , a 4×4 projective transformation ($\mathbf{H}$) is estimated between the camera and the Kinect estimates:

$$\mathbf{X}_k = \mathbf{H}_{4\times4}\mathbf{X}_c \tag{3.1}$$

where $\mathbf{X}_c$ and $\mathbf{X}_k$ are the homogeneous camera and Kinect 3D coordinates respectively. $\mathbf{H}$ is estimated with enough point correspondences and is applied to all the camera 3D estimates to get accurate reconstructions. This is being proposed without any proof here. Results are tabulated in table 3.2 for measurements of the box shown in figure 3.6a.

Distance (mm)	Physical Object	Refined Kinect Scan	Scan with low baseline	Scan with the proposed correction
d_1	190	189.8	182.3	190.882
d_2	129	129.34	123.21	129.73
d_3	63	63.49	61.06	63.1

TABLE 3.2: Improvement in accuracy with the proposed method.

3.2.2.2 Number of Patterns

For fast estimation of 3D models, the image acquisition time should be less. With sinusoidal phase shifted patterns, at least three phase shifts for each frequency are required. But, a non-linear radiometric behaviour known as gamma distortion is present in most digital projectors and cameras. This introduces errors in output intensities leading to error in recovered phase maps and hence the correspondences. The output luminance of a digital projector is then related to input gray level as:

$$L_{op} = L_{max}(\frac{I_{ip}}{I_{max}})^{\gamma} \tag{3.2}$$

where I_{max} is the maximum possible input gray level, L_{max} is the maximum possible output luminance and γ is the gamma distortion (different from the skewness factor in equation 1.6) which is generally greater than 1. Since the accuracy of estimated phase depends on the correctness of the observed intensities, an error is introduced into the recovered phase due to gamma distortion. Figure 3.7 shows the scan of a plane with four phase shifts per frequency. The effects of gamma distortion can be clearly observed.

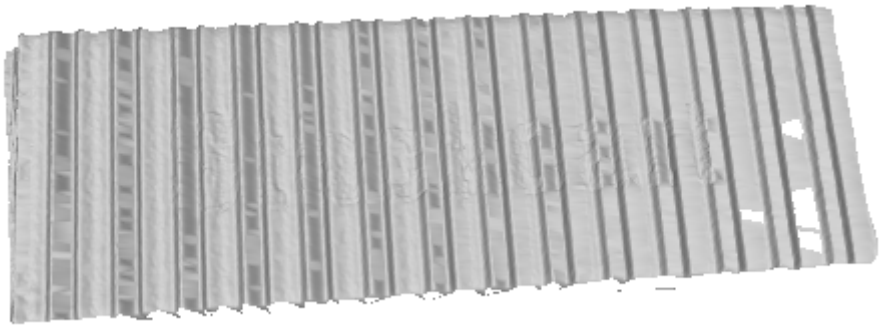


FIGURE 3.7: Scan of a plane with four phase shifts for each frequency.

Dhillon and Govindu [2015] have proposed a radiometric estimation method in their paper on geometric calibration and radiometric estimation in structured light systems. They obtain a phase map of a reference plane placed at an arbitrary orientation. The unwrapped phase map (Φ_c) is mapped on to the projector plane (say Φ_{cp}) by using the homography between them. An error map (Φ_{err}) between the projector phase map (Φ_p) and Φ_{cp} is calculated. Φ_{err} is the error due to radiometric distortion and geometric errors. The geometric error is eliminated by using a moving average filter. They model the final error with respect to the phase values using cubic spline and use this on the phase maps of their scans to eliminate errors due to gamma distortion.

Their approach has been attempted here. Figure 3.8 shows the application of their method on a plain white board. As can be observed, the ripples have been significantly reduced.

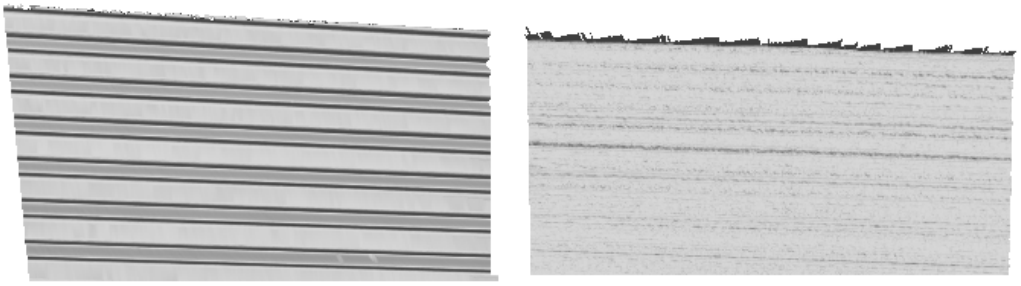
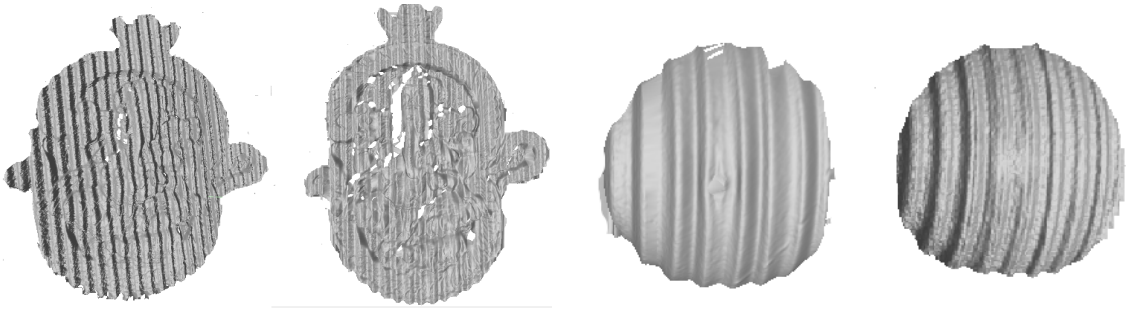


FIGURE 3.8: Radiometric Correction from Dhillon and Govindu [2015] applied on the scan of a white board.

Figure 3.9 shows two more scans on which this radiometric correction algorithm has been attempted. In these two cases, the improvement is negligible. We conclude that this method does not work in this implementation because the different scanned objects used here have different reflictivities or albedoes and the cubic splines have been estimated on the basis of a white board. Thus, due to the non-linear nature of the gamma distortion, different materials will have different errors and hence the estimated splines will vary.



(a) Idol of Lord Ganesha (b) A ball

FIGURE 3.9: Failed cases of the radiometric correction method from Dhillon and Govindu [2015].

Therefore, to reduce the number of patterns while keeping the effects of radiometric distortion in check, different number of phase shifts were used for different frequencies. First, since the phase map for coding the vertical direction is used only for calibration and not for 3D estimation, the phase shifts per frequency were kept to four. Second, since most of the details are captured by the patterns with the highest frequency, sixteen phase shifts were used for the highest frequency (32 cycles) while

four phase shifts were used for the lower frequencies (1 to 16 cycles) for coding the horizontal direction. Hence, number of patterns were reduced from 96 to 60. Figure 3.10 shows the effect on the scans due to change in number of patterns. Scans with 96 and 60 patterns do not have radiometric distortion.

(a) With 48 patterns (b) With 96 shifts (c) With 60 shifts

FIGURE 3.10: Scans depicting the effect of different number of patterns.

3.2.2.3 Computation Time

RGBD cameras like Microsoft Kinect and ASUS Xtion can produce 3D models in video frame rates. But, it should be noted that these systems are calibrated before being used for scanning. In this implementation, due to multiple non-linear optimizations computed inside RANSAC for the calibration, the computation time is slightly high. With careful tuning of the RANSAC algorithm, the system, with the phase and depth maps as input, takes up to five minutes for calibration and 3D model estimation.

Chapter 4

Conclusion and Future Work

A structured light 3D scanner which requires minimum manual effort for calibration and estimates high quality, dense scans has been introduced in this report. By exploiting the accurate calibration of Kinect and the concept of structured light, a new approach of geometric calibration of SLS has been developed. It is to be noted that though the calibration uses standard practices like DLT, non-linear optimization, RANSAC, etc., the proposed idea of calibrating a SLS using Kinect is novel. The method does not require any calibration rigs or printed planar patterns for calibration. The calibration can be done by scanning non-planar objects. It is to be noted that while scanning objects with low depth variation, some objects with more depth variation should be kept alongside to increase calibration accuracy. While RGBD cameras like Microsoft Kinect are relatively cheap and fast, the estimated 3D models are not as detailed. There are structured light 3D scanners like Metron E by Metron 3D for industrial purposes which produce high quality scans within seconds but are relatively costly. The structured light 3D scanner proposed here strikes a good balance between cost and speed.

While the proposed implementation of a structured light 3D scanner has its advantages, there are a few disadvantages as enumerated below:

- Because of the RGBD camera, portability can be an issue. It requires a power source as well as a computer for data acquisition.
- If the ambient light is too bright, the patterns may not be observed properly by the camera.
- The objects to be scanned should be prepared for scanning, that is, they should be as matte as possible. With glossy or specular surface unwanted artefacts may appear in the scans.
- It can be observed in figure 3.3a how the lettering ("Stackcart") on the scanned box appears in the scan and seems like some depth variation. Since the text is in black, light should not have been reflected and some hollow structures in the shape of the letters should have been observed. Also, since the whole surface is plane, a planar scan should be expected. But, neither is the case.

Some areas of improvement which can be undertaken for future work are:

- Automating the whole scanning procedure, from image acquisition to rendering the 3D model.
- Radiometric correction. The attempted radiometric correction did not work as expected. A better, more general correction algorithm should be developed.
- Proof of the projective transformation based approach to reduce occlusion while maintaining the accuracy.

Appendix A

RQ Decomposition

Given a 3×3 matrix $\mathbf{A}$, it is required to compute its RQ decompostion. Let

$$\mathbf{P} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \tag{A.1}$$

$\mathbf{AP}$ reverses the order of the columns of A while $\mathbf{PA}$ reverses the order of the rows. Also $\mathbf{P}^T = \mathbf{P}$, and $\mathbf{PP} = \mathbf{I}_{3\times3}$. So, $\mathbf{P}^{-1} = \mathbf{P} = \mathbf{P}^T$, or $\mathbf{P}$ is orthogonal.

- Reverse the rows of $\mathbf{A}$ i.e, compute $\mathbf{A}_1 = \mathbf{PA}$.
- Compute the QR decomposition of $\mathbf{A}_1^T$ i.e, $\mathbf{Q}_1\mathbf{R}_1 = \mathbf{A}_1^T$.
- Set $\mathbf{Q} = \mathbf{PQ}_1^T$.
- Set $\mathbf{R} = \mathbf{PR}_1^T\mathbf{P}$.

Altogether $\mathbf{R}$ and $\mathbf{Q}$ yield the required decomposition of $\mathbf{A}$:

$$\mathbf{RQ} = (\mathbf{PR}_1^T\mathbf{P})(\mathbf{PQ}_1^T) = (\mathbf{PR}_1^T\mathbf{IQ}_1^T) = \mathbf{P}(\mathbf{Q}_1\mathbf{R}_1)^T = \mathbf{P}(\mathbf{A}_1^T)^T = \mathbf{PPA} = \mathbf{A} \tag{A.2}$$

Appendix B

Normalization for DLT

The projection matrix $\mathbf{P}$ can be estimated using DLT with the below equation:

$$\begin{bmatrix} \mathbf{X}^T & \mathbf{0}_{1\times4} & -x\mathbf{X}^T \\ \mathbf{0}_{1\times4} & \mathbf{X}^T & -y\mathbf{X}^T \end{bmatrix} \begin{bmatrix} \mathbf{P}_{r1} \\ \mathbf{P}_{r2} \\ \mathbf{P}_{r3} \end{bmatrix} = \mathbf{0}_{12\times1} \tag{B.1}$$

$$\implies \mathbf{A} \begin{bmatrix} \mathbf{P}_{r1} \\ \mathbf{P}_{r2} \\ \mathbf{P}_{r3} \end{bmatrix} = \mathbf{0}_{12\times1} \tag{B.2}$$

$\mathbf{P}$ is the right null vector of $\mathbf{A}$ and can be estimated by computing its SVD. The matrix $\mathbf{A}$ contains the pixel coordinates (x,y) and the homogeneous 3D coordinates (X,Y,Z,1). The variation in the pixel coordinates is quite large. Also in the homogeneous 3D coordinates there is a 1, while the other coordinates can have large values (about 1000 mm). These variations can make the matrix $\mathbf{A}^T\mathbf{A}$ ill-conditioned which will result in numerical errors.

The conditioning of the matrix $\mathbf{A}^T\mathbf{A}$ can be greatly improved by normalizing the coordinates such that their centroid is at the origin and they are scaled such that their value is roughly 1.

In the above case, the pixel coordinates can be changed by applying the below

normalization mapping:

$$\mathbf{T} = \begin{bmatrix} s_1 & 0 & -s_1\bar{x} \\ 0 & s_1 & -s_1\bar{y} \\ 0 & 0 & 1 \end{bmatrix} \tag{B.3}$$

where $(\bar{x},\bar{y})$ is the mean of the pixel coordinates. This normalization first translates the coordinates by $(-\bar{x},-\bar{y})$ and then scales them by the factor s_1. The 3D coordinates can be similarly normalized with the below normalization mapping:

$$\mathbf{U} = \begin{bmatrix} s_2 & 0 & 0 & -s_2\bar{X} \\ 0 & s_2 & 0 & -s_2\bar{Y} \\ 0 & 0 & s_2 & -s_2\bar{Z} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{B.4}$$

where $(\bar{X},\bar{Y},\bar{Z})$ is the mean of the 3D coordinates. The scales s_1 and s_2 should be appropriately chosen. Generally, the pixel coordinates are scaled such that their mean distance from the origin is $\sqrt{2}$ and the 3D coordinates are scaled such that their mean distance from the origin is $\sqrt{3}$.

Once, a projection matrix ($\mathbf{P}_1$) with the normalized correspondences is estimated, it is "de-normalized" as:

$$\mathbf{P} = \mathbf{T}^{-1}\mathbf{P}_1\mathbf{U} \tag{B.5}$$

Bibliography

Jason Geng. Structured-light 3d surface imaging: a tutorial. *Adv. Opt. Photon.*, 3 (2):128–160, Jun 2011. doi: 10.1364/AOP.3.000128. URL `http://aop.osa.org/abstract.cfm?URI=aop-3-2-128`.

Daljit Singh Dhillon and Venu Madhav Govindu. Geometric and radiometric estimation in a structured-light 3d scanner. *Machine Vision and Applications*, 26 (2):339–352, Apr 2015. ISSN 1432-1769. doi: 10.1007/s00138-015-0667-0. URL `https://doi.org/10.1007/s00138-015-0667-0`.

Joaquim Salvi, Jordi Pagès, and Joan Batlle. Pattern codification strategies in structured light systems. *Pattern Recognition*, 37(4):827 – 849, 2004. ISSN 0031-3203. doi: https://doi.org/10.1016/j.patcog.2003.10.002. URL `http://www.sciencedirect.com/science/article/pii/S0031320303003303`.

Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, Nov 2000. ISSN 0162-8828. doi: 10.1109/34.888718.

Kenji Hata and Silvio Savarese. Cs231a course notes 1: Camera models.

Martin A. Fischler and Robert C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, 1981. ISSN 0001-0782. doi: 10.1145/358669.358692. URL `http://doi.acm.org/10.1145/358669.358692`.

Paolo Cignoni, Marco Callieri, Massimiliano Corsini, Matteo Dellepiane, Fabio Ganovelli, and Guido Ranzuglia. MeshLab: an Open-Source Mesh Processing Tool. In Vittorio Scarano, Rosario De Chiara, and Ugo Erra, editors, *Eurographics Italian Chapter Conference*. The Eurographics Association, 2008. ISBN 978-3-905673-68-5. doi: 10.2312/LocalChapterEvents/ItalChap/ItalianChapConf2008/129-136.

Sk. Mohammadul Haque, Avishek Chatterjee, and Venu Madhav Govindu. High quality photometric reconstruction using a depth camera. In *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition*, CVPR '14, pages 2283–2290, Washington, DC, USA, 2014. IEEE Computer Society. ISBN 978-1-4799-5118-5. doi: 10.1109/CVPR.2014.292. URL `https://doi.org/10.1109/CVPR.2014.292`.