Index

Chapter:Section	Title	Page number
0:1	Introduction	
0:2	Applications & motivation for this work	
0:3	Structure of thesis	
0:4	Graphical layout of system	
1:1	Correspondence problem:Overview	
1:1:1	Current approaches	
1:1:2	Approach followed in this work	
1:2	System calibration:Overview	
1:2:1	Camera model	
1:2:2	Current calibration methods	
1:2:3	Approach followed in this work	
1:3	Triangulation:Overview	
1:3:1	Current methods to solve triangulation problem	
1:3:2	Approach followed in this work	
2	System calibration	
2:1	System calibration:system setup details	
2:2	Camera calibration	
2:2:1	Procedure for camera calibration	
2:2:2	Visualization of calibration results	
2:3	Projector calibration	
2:3:1	Procedure for projector calibration	
2:3:2	Visualization of calibration results	
2:4	Camera-projector extrinsic calibration	
2:4:1	Procedure	
2:4:2	Visualization of results	

2:5	Evaluation of accuracy of system calibration	
2:5:1	Experiment	
2:5:2	Results	
2:5:3	Visualization	
3	Correspondence problem	
3:1	Overview	
3:2	Procedure	
3:2:1	Pattern generation	
3:2:2	Pattern projection & capture	
3:2:3	Wrapped phase computation	
3:2:4	Unwrapped phase computation	
3:2:5	Absolute phase computation	
3:3	Experiment	
3:4	Result	
4	Triangulation	
4:1	Overview	
4:2	Procedure	
4:3	Visualization of point clouds	
4:4	Experiment	
4:5	Results	
5	Conclusion,observations & proposed future work	
Appendix-I	Measurement accuracy comparision with Microsoft Kinect	
A:1	Kinect:Description	
A:2	Experiment	
A:3	Results	
A:4	Conclusion	
	Bibiliography	

	Figure sources	

Chapter 0

0.1.Introduction
0.2.Applications
0.3.Structure of thesis

0.1.Introduction:

A 3D scanner is a combination of hardware & software components used for acquisition of three dimensional information from a scene. Recent applications[**1**] of 3D scanners seems to suggest that they are not only used for metric reconstructions but also for acquiring information regarding relative placement of objects in 3D space which do not essentially requires metric information examples are a*ffine* and *projective* reconstructions[**2,3**] example application include robotic navigation where 3D coordinates in terms of physical units are not required but only a non-metric 3D information suffices[**4**],whereas the *euclidean* reconstruction comes under category of *metric* reconstruction and is also the subject of this work. Although recent work on Monocular 3D[**6**] has lead to development of methods which can extract approximate three dimensional information even from single view using cues like perspective distortion,boundary shades,shadow,occlusion etc,exact metric reconstruction still requires at least 2 views. In simplest case with 2 views,there is a need to recover a relationship between these views so as to compute depth of a point common to both views with respect to a global coordinate system. But since depth is to be interpreted and utilized for real world practical applications(as is the case with this work) it has to be expressed in terms of physical units like 'mm','cm','feet' etc. So clearly above discussion point towards three major problems associated with 3D metric reconstruction:Correspondence between views,Calibration of viewing elements to determine the relation between abstract world of these elements(in our case these elements are projector & camera) and our physical world,and computation of depth given correspondence & calibration information. Specifically for 3D reconstruction,first system is calibrated,then correspondence between camera & projector is calculated,using the calibration and correspondence information triangulation is done to compute depth.

0.2.Applications & objective of this work:

3D scanners find applications in robotic controls for example a laser scanner can act like an eye for robot,in site modeling and lay-outing,establishing benchmark for preexisting shape/state for studying structural changes in response to extreme environmental conditions,to create Geometric information systems,it is also used for sub-surface scanning in mines. Of course in entertainment where currently Microsoft Kinect is a leading example. It is also used in reverse-engineering a mechanical component which requires a precise digital model of object. In cultural heritage preservation,where Michelangelo,Monticello,Kasubi-tombs etc are prime examples. It is also used in Medical CAD/CAM for example to produce orthosis,prosthesis and dental implants. In industrial quality inspection and meteorology it is used for measuring geometric dimension accuracy,automation of quality assurance process for example in assembling a car it must be ensured that geometry of individual components is accurate so that it can fit together for reliable operation.

In this work,a 3D scanner based on a structured light technique is developed and its accuracy is analyzed to determine various factors which affect final three dimensional measurements using this scanner. Motivation for this work comes from the need of acquiring expertise in the field of 3D scanning to be able to procure commercial 3D scanners that are used in Hospitals for dental scanning and other purposes. In addition,CFD applications on which Computer graphics section is working,an extension of our work can be used to generate 3D models of real surface which can then be used

to perform simulations.

0.3.Structure of thesis:

Chapter-1 describes the current state-of-art for solving correspondence problem and brief introduction to approach used in this work,for system calibration it describes most frequently used system calibration approaches and brief outline of algorithms used in this work for triangulation it describes the current approaches followed to compute depth given correspondence & calibration information and basic introduction to the approach followed in this work. Chapter-2 describes the details of development of system calibration module and experiments performed to evaluate its accuracy. Chapter-3 describes the details of correspondence computation module and experiments performed to evaluate its accuracy. Chapter-4 starts with a brief recap of theory of triangulation used in this work followed by description of procedure used for verifying its accuracy and its results. Chapter-5 discusses our observations from this work followed by proposed future works. Appendix describes the experimental comparison with Microsoft Kinect to which serves as an example to assess accuracy of our system with respect to state-of-art 3D scanner devices. Bibliography & reference to sources of figures concludes this thesis.

0.4.Graphical layout of the system developed:

Here we graphically represent the layout of our 3D scanner system which can be referred during thesis.

Chapter-2 is concerned with the development of *System calibration module*,Chapter-3 is concerned with development of *Pattern generator,pattern projection capture,phase wrapping* and *phase unwrapping*,and *absolute phase computation* modules,Chapter-4 is concerned with development of *Triangulation module*.

CHAPTER 1

Structure of chapter:

1.1.Overview of correspondence problem
1.1.1.Current approaches
1.1.2.Approach followed in this work

1.2.Overview of system calibration
1.2.1.Camera model
1.2.2.Current calibration techniques
1.2.3.Approach followed in this work

1.3.Overview of triangulation
1.3.1.Current triangulation methods
1.3.2.Approach followed in this work

1.1.1.Overview of correspondence problem:

As described earlier,correspondence problem involves determining a relationship between 2 views but since these views come from projective elements like cameras,projectors,a pixel-to-pixel relation between this pair(called *binocular-pair or stereo-pair*) must be determined. In nutshell,establishing correspondence(technically called *stereo-correspondence*) means to know for a point in 3D world,which pixel in camera & in projector(or another camera) will see it. Following section gives introduction to the current state-of-art in field of stereo-correspondence.

1.1.2.Current approaches:

To solve correspondence problem following classification is reported in literature[**6,7**]:

1.Contact techniques:
- 1.1.Non-destructive techniques:
  - 1.1.1.Coordinate measurement machines
  - 1.1.2.Jointed arms
- 1.2.Destructive techniques:
  - 1.2.1.Slicing

2.Non-contact techniques:
- 2.1.Transmissive techniques:
  - 2.1.1.Industrial CT
- 2.2.Reflective techniques:
  - 2.2.1.Optical techniques:
    - 2.2.1.1.Active techniques:
      - 2.2.1.1.1.***Active stereo***
      - 2.2.1.1.2.Imaging radar
      - 2.2.1.1.3.Active depth from defocus
    - 2.2.1.2.Passive techniques:
      - 2.2.1.2.1.Passive stereo
      - 2.2.1.2.2.Depth from defocus
      - 2.2.1.2.3.Shape from shading
      - 2.2.1.2.4.Shape from silhouette
  - 2.2.2.Non-optical methods:
    - 2.2.2.1.Microwave radar
    - 2.2.2.2.SONAR

In this work we have focused on active stereo approach using structured light technique. So our survey will describe current state-of-art in structured-light

techniques. Structured light systems deploy off-the-shelf video cameras & projectors as their binocular elements. Here a camera pair of passive stereo methods is replaced by a camera-projector pair. It provides solution to the problem of correspondence for texture-less surfaces experienced by passive stereo camera pair because of lack of texture marks which act as important cues for establishing correspondence. Projection of explicit patterns which create an *artificial texture* on object surface,greatly simplifies the problem of correspondence. In these techniques,projector projects a pattern(s) which encode the information regarding the projector pixel which has illuminated a point in the scene. Camera captures these patterns and by decoding the pattern which is functionally equivalent to solving the correspondence problem in passive stereo vision,it extracts the unique ID of projector pixel that illuminated the same point in 3D space as is captured by camera. **Figure-1[F.1]** provides conceptual representation of structured-light systems:

Note:Here angles **theta & alpha** are for corresponding pixels in projector and camera. Projected pattern in the figure is just representative of structured-light technique.

Structured light techniques have been classified in 3 major classes[**6**,**7,22**] based on how correspondence information is encoded in the projected patterns:
a).Time multiplexing:Every point is encoded by series of intensities across multiple patterns.
b).Spatial encoding: Every pixel is encoded based on intensities of its neighborhood pixels.
c).Direct encoding:Every pixel is encoded by its unique intensity and does not depend upon any spatial or temporal correlations among pixels.
d).Hybrid encoding:It encodes pixels both spatially and temporally to project lesser number of patterns than conventional time-multiplexing techniques and to have more robustness as compared to spatial methods.

It should be noted however that this classification do not enforce unique code for each projected pixel i.e., there may be a set of pixels with same code. This pattern design decision depends upon required resolution in the reconstructed scene.

a).Time multiplexing

Time multiplexing techniques encode unique ID's for pixel(s) of projector over multiple patterns projected sequentially over a period of time. Camera captures all these pattern in an order to determine correct ID for each pixel(or pixel group).Depending upon type of correspondence required i.e. camera-pixel to projector-pixel or camera pixel plane to projector-pixel number of axes of codification can be chosen to be both vertical and horizontal or one of them respectively. To be more clear,axis of codification means axis of projected pattern along which the codes are embedded other axis will simply have same copy of intensity for each code in each pattern and hence will not contribute to pixel codification.Examples of time-multiplexing based structured light patterns are binary-coded patterns,N-ary codes,gray code+phase shift method. Here we will discuss these methods as representatives for time multiplexing approach.

a.i).Binary coded patterns:

In these patterns**[9]**,any pixel in camera & projector is identified by the sequence of *binary* intensities it receives or projects respectively. Specifically,for a NxM projector resolution,if pixels are to be divided into K groups along say Horizontal axis then a total of N/K code-words will be used which will requires projection of l**$\log_2$(Ceil(N/K))** patterns sequentially,where **ceil(x)** represents largest integer lesser than or equal to x . A pattern say **j**th will represent intensities corresponding to the **j**th bit of all code-words across its horizontal axis. Hence every pattern is called a bit plane. **Figure-2[F.2]** shows projection of simple binary-coded patterns scheme and **figure-3[F.1]** shows design of code-words**:**

After all patterns are captured by camera,a code at every pixel is computed based on intensities incident on that pixel during pattern projection. A single pattern determines single bit in code-word of all pixels in the camera.

a.ii).N-ary coded patterns:

In case of binary-coded patterns it can be observed that for high resolution more code-words will be required which directly increases the number of patterns to be projected. For applications requiring faster acquisition rates,number of patterns can be reduced yet meeting the resolution requirement if we increase number of intensity levels that a

pixel can have[**9,10,11**]. This approach can allow for dynamic scene shape recovery at the cost of decreased robustness against environmental noise. As it can be deduced,color-coded patterns are special cases of this technique.**Figure-4[F.1]** shows the gray-level coding scheme,here multiple allowed gray-level can be observed:

a.iii).Combination of gray-coded patterns and phase-shift techniques:
Gray-coded patterns were proposed to increase robustness of code recovery process against possibility of false-code extraction as compared to simple binary-coded patterns,since successive gray code-words differ at only one bit position thereby reducing chances of assigning same code to the pixels belonging to neighboring code-word groups. It has a straightforward analogy with the considerations in code designing for noisy communication channel. For a treatment of structured-light patterns in way similar to communication code designing over a noisy channel,refer**[12]**.In phase-shift technique,phase shifted sinusoidal interference fringes are projected on object surface. In camera unique code to each pixel is assigned in the form of phase of the signal incident on it. Similarly in projector image a phase value is assigned to each pixel which represents spatial position of that pixel in the projected sinusoidal periodic waveform. It can be deduced that this phase value indirectly encodes the position of projector pixel that projected that signal and hence the correspondence information. It should be noticed that gray-coded patterns provide a way to determine a region of pixels uniquely by assigning unique code to it,whereas phase-shift technique can provide phase variation up to pixel-level within that region. Although phase-shift technique is potent to be used alone to provide pixel-level resolution but computational complexity of accurate phase value computation constraints the image resolution & scene geometry. Interested reader may refer [**13**] for details of problems in accurate phase estimation called *phase-unwrapping.* Hence to alleviate problem of low resolution of gray-coded patterns and computationally expensive phase-unwrapping problem,combination of both techniques is used. Here period number of a sinusoidal fringe is determined with the help of gray-coded pattern and pixel-level correspondence within that period is computed using phase-shift technique. Although it comes at a cost of increased number of patterns(typically greater than 10) that must be projected as compared to the original phase shift approach which typically requires 3 or 4 patterns. A variant of this approach is the subject of our work.

Figure-5[F.1] describes this approach:

It should be noted that in pattern design,width of a code-word(in pixels) is equal to width of a period in phase-shifted pattern to assign a unique code-word to each fringe period and hence perform phase unwrapping.

b).Spatial encoding methods:

As already mentioned these methods extract more information per pattern by encoding unique identity of pixels in their neighborhood(called *window*) also. Although it can be clearly argued that this approach will have less robustness as compared to time-multiplexing methods since error in a decoding of one pixel will affect decoding of all dependent pixels. Here size of the window is directly proportional to the number of pixels encoded and inversely proportional to the number of intensity levels used for encoding. Basically,aim of these methods is to obtain correspondence from single captured image which facilitates them to be used for dynamic scenes(like moving objects).As can be noticed,increased number of allowed intensity-levels increasing chances of error in decoding because natural scene illumination can potentially interact with these projected patterns making accurate pattern decoding more challenging than binary-coded patterns. Here we discuss De Bruijn patterns and M-arrays as representatives of this technique.

b.1).De Bruijn sequence based pattern:

De Bruijn sequence**[14]** of order 'm' over an alphabet of 'n' symbols is a pseudo-random sequence of length $\mathbf{n^m}$ where every possible combination of length 'm' occurs only once. These sequences have been adapted for pattern generation by considering number of allowed intensity-level as 'n' and window length as 'm'.In terminology of graph theory it can be interpreted as a graph of n^m vertices and problem of generating debruijn sequence is to determine a Hamiltonian path in that graph,this problem is also called *Traveling salesmen problem*. Specifically,a pattern encodes a debruijn sequence. Pattern is divided into slits where each slit has width equal to the window size of pattern. Intensities of sub-slits are generated using individual code-words occurring in debruijn sequence. To decode a pixel it is only required to determine the code-word or window to which it belongs which will depend upon the intensities of its neighborhood pixels.

Figure-6[F.2] shows patterns generated using debruijn sequence of various orders and window size:

(a) orde: 4, window: 3

(b) orde: 4, window: 3

(c) orde: 3, window: 4

(d) orde: 4, window: 4

It can be observed that the debruijn sequences encoded in these patterns are one-dimensional. Hence they will allow for projector pixel to camera pixel plane mapping but not for camera-pixel to projector-pixel mapping. Hence final triangulation can be performed using ray-plane intersection.

b.2).M-array based pattern:

M-arrays[15,16] are bi-dimensional extensions of debruijn sequences discussed in section 2.1.Hence a M-array of NXM dimension will have all 2D code-words(in window) of dimension WXH that will occur only once in complete sequence/array. One code-word will be present in one window,so it precisely means that one window will appear only once in the whole array.

Figure-7[F.3] shows a M-array pattern:

c).Direct coding techniques:

In this technique,every encoded pixel is identified by its own color/intensity**.** Although theoretically high resolution for dynamic scenes can be attained but practically this approach is highly susceptible to noise because of large number of intensity-levels used. It also implies that camera with very large pixel depth have to be used for

detecting changes in intensity across the scene to extract code from it. Efforts in this direction include grey patterns**[17,18,19]** for example intensity ratio based grey pattern by **[17]** which fades away from black to white. Colored patterns[**20,21**] for example **[20]** proposed projection of rainbow pattern whereas **[21]** proposed relatively robust patterns which effectively cancel out the object color by projection of three shifted patterns but at the cost of increasing acquisition time. This approach can be classified as hybrid approach also since it includes combination of direct and temporal techniques. Figure**-8[F.3]** shows the 3 shifted rainbow pattern as representatives of colored patterns used for direct coding techniques:

d).Hybrid coding techniques:

In order to attain benefits of both spatial & temporal techniques,hybrid patterns**[23]** were also proposed where patterns are projected temporally but their encoding also depends on neighborhood thereby allowing more information to be encoded per pattern and increased robustness from temporal technique since multiple frames contain the encoding information hence dependence on neighborhood pixels is less as compared to purely spatial neighborhood approach. This approach allowed for slow motion in the scene which is exceptional from point of view temporal-encoding schemes.

1.1.3.Approach followed in this work:

In this work,we have attempted to investigate the accuracy achievable by Phase-shift technique for 3D scanning since currently it is the most popular approach**[24,25]** for its claimed capability to achieve high accuracy and resolution in the structured-light category. We decided to develop it also rather than just accuracy analysis because there is no open-source implementation available for this technique except the work of **[26]** but it does not have documentation,is still under development and it is only for windows platform whereas we are working mainly on Linux systems. Work by **[27]** also attempted for three phase scanning but at the time of development had no means for system calibration and had only naive phase unwrapping algorithm which was far from being usable as a reference for accuracy analysis. Although originally we planned and fully developed the approach purely based on phase-shift technique but our phase-unwrapping implementation[**28**] was not converging to a solution and was running for >3hrs without solution(**on Core2Duo processor E8400 at 3.0GHZ and 2 GB RAM**) for an image resolution of 960X720 even after several attempts to check correctness of implementation hence keeping project deadline & time-efficiency of solution to be developed under consideration we shifted to a combination *phase-shift & binary-coded technique* which is similar to the technique described in **Section-a.iii**(except that binary codes are used instead of grey codes) to reduce the computation time requirement in the phase-unwrapping process. Although after this work we will be working on our previous implementation to improve its running time

efficiency. In addition we have observed some deficiencies of the hybrid approach used in this work which we will discuss in chapter-5.Approach is step-wise explained in chapter-3.

1.2.System calibration:Overview

For performing non-metric reconstruction only point-to-point correspondences between camera & projector are needed to perform triangulation[**3**]. But to relate the depth computed using triangulation with physical metrics such as 'mm','cm' etc there is need to relate the camera & projector geometry with physical measurement units,this task is accomplished by process of System calibration. Calibration is a process of determining intrinsic & extrinsic characteristics of the imaging devices(camera & projector in our case).Intrinsic characteristics include focal length,coordinates of principal points of lens used in camera & projectors,image axis skew coefficients and lens distortion coefficients. Extrinsic characteristics include rigid body transformation that can transform a point represented in World-coordinate system to individual coordinates systems of camera & projector. In next section,we will describe camera model which describes clearly the problem of calibration,following which we will describe the current approaches to solve this problem and approach used in this work. In the end,we mention a note on relative extrinsic calibration of projector-camera system,which is a process of determining relative orientation and position of camera with respect to projector.

1.2.1.Camera model:

Camera model is a concise mathematical abstraction of the internal(its optics and sensor arrangement) & external(its position relative to a scene) geometry of camera which is responsible for the imaging process. In this section we will describe these geometric parameters and hence the imaging process. Camera model described here is referred from seminal paper by **[29]** other calibration techniques also follow variants of this formulation. In its original formulation,**[29]** described it in form of a sequential process i.e.,in form of steps and we will follow that approach here. Relatively recent work in **[30]** has shown projector to be *inverse-camera* hence further discussion on camera calibration holds equally to projector as well. Chapter-2 will describe how to calibrate projector similar to camera with real example.First we describe the coordinate-systems which are relevant for geometric modeling of imaging process under consideration:

<u>World-coordinate system:</u>
World-coordinate system defines the reference frame in the real world that will be used for defining the positions of object in scene in physical units. Typically any location in the real scene can be used as world coordinate system axes. For example in our work we assume world coordinate system origin to be present at a corner of a physical board.

<u>Camera-coordinate system:</u>
This is a 3D reference coordinate-system internal to camera,it is similar to world coordinate system since it also works in physical units. Its origin is assumed to be at center of projection of camera. It is used to generate 3D interpretation of the scene from camera viewpoint.

<u>Image-coordinate system:</u>
This is a 2D reference coordinate-system for indexing points in the real image. Note here *real* means that it can take real values as coordinates. It is used to index the points in perspective projection of 3D world.

Computer-frame coordinate system:
This is a 2D reference coordinate-system which indexes pixels in the computer-frame. It can only have integral coordinates.

Figure-9[F.4] illustrates the coordinate systems except computer-frame coordinate system since it is not *geometrically* connected with other coordinate systems:

Here (X_w, Y_w, Z_w) are coordinates of P in world-coordinate system, (x, y, z) are corresponding coordinates in camera-coordinate system, (x_u, y_u) are its undistorted coordinates in real-image plane, (x_d, y_d) are its distorted coordinates(explained in Step-3 below).

Step-1:
Rigid body transformation from object world-coordinate system to camera coordinate system:
Points in the world-coordinate coordinates system (X_w, Y_w, Z_w) are transformed to the camera-coordinate system (X_c, Y_c, Z_c) . To model this process rotation(R) and translation(T) matrices are used as:

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = R * \begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} + T$$

$$where\ R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}, T = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix}$$

Rotation & translation parameters are called *extrinsic* parameters of camera/projector. It should be noted here that rotation followed by translation is used since it ensures unique transformations,whereas translation followed by rotation can result in multiple possible transformations**[29]**.

Step2:

Perspective transformation from camera-coordinates to real image-coordinates:

In this step,3D scene as viewed from camera-coordinate origin is perspective projected to a plane. Hence it is a 3D to 2D transformation resulting in loss of Z-coordinates(i.e.,depth) since multiple points are projected onto same point in the image-plane. Following equations describe this process:

$$X_u = f\left(\frac{X_c}{Z_c}\right)$$

$$Y_u = f\left(\frac{Y_c}{Z_c}\right)$$

$$where\ 'f'\ is\ camera\ focal\ length, (X_u, Y_u)\ is\ perspective\ projection\ of\ (X_c, Y_c, Z_c)\ on\ real\ image\ plane$$

Step-3:

Radial lens distortion:

In this step,real-image coordinates are distorted due to imperfections in practical lens & relative geometry of lens & sensor-plane. Although Tsai's work only accounted for radial distortions subsequent works have accounted for tangential**[31]** & prism distortions[**38**] also in an attempt to increase the accuracy of estimated calibration parameters further. Following equations account for the effect of radial distortion:

$$X_d + D_x = X_u$$

$$Y_d + D_y = Y_u$$

$$where\ (D_x, D_y)\ denotes\ radial\ distortion, (X_d, Y_d)\ denotes\ distorted\ image\ plane\ coordinates$$

$$D_x = X_d(1 + K_1 * r^2 + K_2 * r^4 + K_3 * r^6 + ...)$$

$$D_y = Y_d(1 + K_1 * r^2 + K_2 * r^4 + K_3 * r^6 + ...)$$

$$where\ r = \sqrt[2]{X_d^2 + Y_d^2}\ denotes\ the\ radial\ distance\ from\ the\ center\ of\ lens$$

Step-4:

Transforming real-image coordinates to computer-frame coordinates:

Finally to map 3D world-coordinates to frame coordinates of camera,i.e.,the pixels that we see,following equations are assumed:

$$X_f = s_x * (d_x')^{-1} * X_d + C_x$$
$$Y_f = d_y^{-1} * Y_d + C_y$$
$$where,$$
$$(X_f, Y_f): computer\ frame - buffer\ coordinates\ of\ (X_d, Y_d)$$
$$s_x: uncertaintiy\ image\ scale\ factor$$
$$d_x: center - \%to - center\ distance\ bewteen\ sensor\ cells\ along\ X - direction$$
$$d_y: center - \%to - center\ distance\ between\ sensor\ cells\ along\ Y - direction$$
$$N_{cx}: Number\ of\ sensors\ along\ X - direction$$
$$N_{fx}: Number\ of\ pixels\ along\ an\ X\ line\ sampled\ by\ computer$$
$$d_x' = d_x\left(\frac{N_{cx}}{N_{fx}}\right)$$

It should be noted that at full camera resolution there is one-to-one relation between number of sensor elements and number of pixels in computer-frame buffer hence $N_{cx} = N_{fx}$ and hence $d_x' = d_x$ but in general $dx' >= d_x$.[29] attributed s_x to be due to possibility of mismatch between camera image acquisition hardware and camera scanning hardware.

Figure-10 [F.4] illustrates the complete imaging process:

So based on above discussion we can deduce that to completely define the geometric imaging process of camera we have to estimate following parameters:
1).Internal parameters:
    1.1).Camera lens focal length
    1.2).Position of principal point on the real-image plane
    1.3).Lens distortion parameters
2).External parameters:
    2.1).Rotation between camera-coordinate system & world-coordinate system
    2.2).Translation between camera-coordinate system & world-coordinate system

1.2.2.Current calibration techniques:

A taxonomy was proposed by **[31]** for camera calibration approaches based on their requirement of additional calibration objects as:

a).Photogrammetric calibration:

These techniques perform camera calibration by observing a calibration object. Calibration object can be any object with *known* geometry i.e.,we know world-coordinates of feature-points on the object that we want to use for calibration. In this category many techniques for both 3D & 2D calibration objects were proposed. For example use of 3D calibration object with three orthogonal planes or planer calibration rigs with known translation across its views have been used for calibration**[29]** although requiring elaborate setup for calibration & often time-consuming.

b).Self-calibration:

These techniques are more flexible than photogrammetric techniques in that they do not require an explicit calibration rig for calibration,but by moving the camera across a static rigid scene they determine constraints on internal parameters of camera that can be used for calibration.**[32][33]**

Other than these techniques there are techniques that do not fall into these categories such as camera calibration using vanishing points for orthogonal directions[**34**],calibration from pure rotation[**35**].

In this work we have used photogrammetric method for camera & projector calibration. We describe brief details of following methods since they are highly cited and practically used algorithms:
a).R.Y.Tsai camera calibration algorithm:

b).Z.Zhang camera calibration algorithm

c).J.Heikkila et al. camera calibration algorithm

1.2.3.Approach followed in this work:

For camera & projector calibration we have used an extension of **[31]** method first described & developed in **[36]** & later ported to OpenCV. Unfortunately,author in **[36]** has not published the details of algorithm except describing the approach to be based on camera model described in **[37]** and implementation as a variant of **[31]** hence OpenCV documentation other than reading the source code itself is the only source of theoretical information on this technique. During this project work,an effort is initiated to fully understand the calibration algorithms and effect of each calibration parameters on accuracy of 3D reconstruction by actually implementing an algorithm

so that every input to the algorithm can be changed and its effect seen. For which we selected and partially implemented technique described in **[37].**We propose work in this direction in chapter-5.Here we present brief outline of camera/projector calibration algorithm implemented in OpenCV/Matlab**[36][42]**:

a).A planer calibration board is selected with known world-coordinates of reference points on it.

b).Calibration board is shown for multiple views(at least 2) in front of camera by varying its rotation and translation(amount need not be known) with respect to camera and for every pose feature-points are detected in camera.

c).After all such views are captured & feature-points detected,homography for each view is calculated.

d).Closed form solution of intrinsic parameters from homograpies using orthogonality of vanishing points is computed but unlike **[31]** no initial estimates of distortion coefficients are computed.

e).Initial estimates computed earlier are given as input to Maximum likelihood estimator Levenberg-Marquardt algorithm to refine the parameters.

<u>Algorithm for extrinsic calibration of camera and projector:</u>

Extrinsic parameters(Refer Step-1 in camera model,Section-1.2.1) relate camera-coordinate system & projector-coordinate system to world-coordinate system. For extrinsic calibration,OpenCV**[44]** performs minimization of re-projection error between observed point projections given as input(i.e.,image points) and projection computed by applying the estimated model to the given object points. Extrinsic parameters that minimize this criteria are considered as the optimal pose(Rotation & translation) parameters.

<u>A note on Relative calibration of projector & camera:</u>

After camera & projector are calibrated extrinsically,there is need to relate their coordinate-systems with each other so that a point in projector coordinate-system can also be expressed in terms of camera-coordinate system. This is required because of an implicit assumption in triangulation that requires intersection of optical rays from camera & projector,which is possible only if we can express both rays in a common coordinate-system either in that of camera or projector. This is typically achieved using our world -coordinate system as a common link between projector & camera. Following equations relate camera-coordinate system to projector-coordinate system via world-coordinate system**[43]**:

$$P_c = R_{wc} * P_w + T_{wc}$$

$$P_p = R_{wp} * P_w + T_{wp}$$

$$hence\ ,$$

$$P_c = \underbrace{(R_c * R_p^{-1})}_{(R\,\mathrm{pc})} * P_p + \underbrace{(T_c - R_c * R_p^{-1}}_{(T\,\mathrm{pc})} * P_p)$$

$$where\ ,$$

$$P_w : A\ point\ in\ world\ coordinate\ system$$

$$P_c : Representation\ of\ P_w\ with\ respect\ to\ camera\ coordinate\ system$$

$$P_p : Representation\ of\ P_w\ with\ respect\ to\ projector\ coordinate\ system$$

$$R_{wc} : Rotation\ transformation\ from\ world\text{-}to\text{-}camera\ coordinate\ system$$

$$R_{wp} : Rotation\ transformation\ from\ world\text{-}to\text{-}projector\ coordinate\ system$$

$$T_{wc} : Translation\ transformation\ from\ world\text{-}to\text{-}camera\ coordinate\ system$$

$$T_{wp} : Translation\ transformation\ from\ world\text{-}to\text{-}projector\ coordinate\ system$$

$$R_{pc} : Rotation\ transoformation\ from\ projector\text{-}to\text{-}camera\text{-}coordinate\ system$$

$$T_{pc} : Translation\ transformation\ from\ projector\text{-}to\text{-}camera\ coordinate\ system$$

1.3.Triangulation:Overview

Triangulation is a technique of computing distance of a point with respect to a known reference coordinate system,given calibration parameters,relative pose & correspondence information between imaging elements by performing optical ray-ray or optical ray-plane intersection between corresponding pixels(or plane of pixels) of imaging elements. When noise is present the rays will not intersect exactly,hence typically an approximate intersection point is estimated under some assumed noise model.

1.3.1.Current techniques for triangulation:

In this section we describe the techniques used for triangulation. This section is largely based on **[4].**In this work only metric reconstruction(specifically,euclidean reconstruction) is attempted but readers interested in more detailed exposition of all types of reconstructions possible given point correspondences between multiple views of the scene,can refer **[39][3][40].**

a).Mid-point method:

This method proposes to use mid-point of common perpendicular between optical rays corresponding to matching points as the intersection point.

b).Quasi-Euclidean approach:

In this method a approximation to the correct euclidean frame is selected and mid-point method is performed in it.**[4]** describes it as a sub-optimal method.

c).Linear triangulation

Let $u_c = P * U_w$,where P is projective transformation between a point u_c in camera image plane and a point U_w in world coordinate system. Here $u_c = w * (x\, y\, 1)^T$ where w is an unknown scale factor and u_c is in homogenous coordinate representation. Let us denote p_i^T as ith row of P.This relation gives us three equations for one view:

$$w * x = p_1^T * U_w$$
$$w * y = p_2^T * U_w$$
$$w = p_3^T * U_w$$

Eliminating w ,we get two constraints on U_w :

$$x * (p_3^T * U_w) = p_1^T * U_w$$
$$y * (p_3^T * U_w) = p_2^T * U_w$$

Hence using two views we get 4 equations in three unknowns which can be represented in form of system of Homogenous linear equations:

$$A * U = 0$$

Linear triangulation methods**[45]** are based on solving system of linear equations mentioned above.

c.i)Linear-Eigen method:

Minimize $\|AU\|$ under the constraint $\|U\| = 1$ .Solution is unit eigenvector,corresponding to smallest eigenvalue of $A^t * A$. This problem may be solved using Singular-value-decomposition or Jacobi's method for finding eigenvalues of symmetric matrix.

c.ii).Linear-LS method:

This method assumed there can be no point on plane at infinity,thereby setting $w=1$ in U_w .This constraint allows system of four linear equations for 3 unknowns whose least-squares solution can be computed using for instance Singular value decomposition method.

d).Iterative linear methods

[4] points out reason for failure of Linear method is due to lack of geometric interpretation of $\|AU\|$ and they do not minimize the optimal cost criteria proposed by them. To account for this an equivalent *iterative* version of methods is proposed in literature called *iterative-LS* & *iterative-Eigen.* Although authors in **[4]** claim iterative methods to be more insensitive to projective transformations than corresponding non-iterative versions. But they also point out that non-convergence is an issue with these methods especially in the case when matching points are near epipoles,hence they suggest these methods to be used with some other supportive method which can be used in case non-convergence is detected.

e).Minimizing the sum-of-magnitudes of distances:

In this method,sum magnitude of Euclidean distance between observed noisy correspondence (u_r, u_l) & possible candidates (u_r', u_l') is minimized ,ideal solution is the one which will exactly satisfy:

$$u_r^T * F * u_l = 0$$

$$\textit{where},$$

$$u_r : \textit{position of a point in right view}$$

$$u_l : \textit{position of corresponding point in left view}$$

$$F : \textit{Fundamental matrix}\ [40]$$

Hence the cost function becomes:

$$c = d(u_r, u_r') + d(u_l, u_l')$$

$$\textit{where},$$

$$d(u_i, u_j)\ \textit{represents the}\ 2\text{D}\ \textit{euclidean distance between two image points}\ u_i\ \&\ u_j$$

f).Polynomial method

In this approach, Gaussian error model is assumed in detected features in image. It minimizes sum of squares of euclidean distances between corresponding ideal & measured points in the individual images as:

$$c = d(u_r, u_r')^2 + d(u_l, u_l')^2$$

It is proved to be optimal provided the assumed Gaussian noise-model holds. **[4]** argues that problem of accurate triangulation is more concerned with accurate determination of point-correspondences. This approach assumes that camera matrices(& hence fundamental matrix) is known more accurately than point-correspondences and assumption that correct matching pair will be most closer to the measured matching pair.

1.3.2.Approach used in this work:

In this work,we actually implemented **[41],**Mid-point method and Linear-LS for triangulation but we found only the Linear-LS approach seems to give relatively accurate results. But we have not analyzed why other approaches do not work since our initial target was to develop a 'system' that can perform 3D reconstruction. We therefore propose this work as our future work in chapter-5.

CHAPTER 2: System calibration

Structure of chapter:

2.1.System calibration:System setup details

2.2.Camera calibration
2.2.1.Procedure used for camera calibration
2.2.2.Visualization of calibration results(mention computed calibration parameters)

2.3.Projector calibration
2.3.1.Procedure for projector calibration
2.3.2.Visualization of projector calibration results(mention computed calibration parameters)

2.4.Camera-projector extrinsic calibration
2.4.1.Procedure
2.4.2.Visualization of results(mention computed calibration parameters)

2.5.Evaluation of accuracy of system calibration results
2.5.1.Experiment
2.5.2.Results & conclusion

2.1.System calibration:Overview

As mentioned and explained in **Section-1.2**,system calibration is composed of camera calibration,projector calibration & relative extrinsic calibration of camera with respect to projector. We developed a module for system calibration which performs above mentioned functions using OpenCV camera calibration & extrinsic calibration algorithms.

System setup:
In this work,we used **Logitech Quickcam sphere AF** web-cam at 1600X1200 pixels resolution,**Sharp PG-F200X** projector at 1024X768 pixels resolution,and 4 markers (shown in figure as 'black' points) as world coordinate references forming a rectangular region of 1900mmX1700mm and a physical checkerboard with square size 25.6mm arranged in a 10X8 square grid.**Figure-11** shows setup of our 3D scanner system:

2.2.Camera calibration:

In this work,we performed camera calibration using OpenCV camera calibration function whose theoretical foundations were laid by the well known algorithms by **[31]** & **[37].**Basic overview of algorithm was presented in **Section-1.2.3**.

2.2.1.Procedure:

Aim of this process is to acquire 3D-2D correspondences between world coordinates system & camera image coordinate system which are then used to estimate intrinsic & extrinsic parameters of camera. Here extrinsic camera parameters include the rotation and translation matrices of World coordinate system with respect to camera coordinate system. World coordinate system origin is assumed to be at top-left corner of the physical checkerboard as shown in **Figure-12:**

Multiple views of the checkerboard are captured by camera and inner corners are detected with sub-pixel accuracy. Each such view provides a set of 3D-2D correspondence and adds a separate set of rotation & translation parameters to the list of unknowns. There are 15 unknowns in the camera model assumed in OpenCV: $f_x, f_y, c_x, c_y, r_x, r_y, r_x, t_x, t_y, t_z, k_1, k_2, k_3, p_1, p_2$ hence at least 15 equations are required to solve for the unknowns. This requires at least 2 views since a single view can provide only 8 constraints(i.e.,4 points are required to completely define a view or plane).Practically,we use more than 2 views(typically greater than 10) with more than 4 points per view to account for noise in corner-detection. Acquires 3D-2D point correspondences are used to calibrate camera as explained in **Section-1.2.3.**

Figure-13 shows some views of checkerboard captured by camera for calibration:

2.2.2.Result visualization:
For visualization of camera calibration process we developed a program to plot the views used for calibration. This visualization is similar to **[36]** but it was not available for projector calibration due to which we developed our own.

Figure-14 shows a snapshot of 3D plot of all views of checkerboard used for camera calibration:

2500
2000
1500
1000
500
0
-300
-200
-100
0
100
200
300
400
500
Y
camera
Y
-800
-333
133
600
X

(a)

(b)

Here,figure-14(a) shows that our calibration range was 50cm to 250cm along Z-axis,-25cm to +45cm along Y-axis. Figure-14(b) shows that calibration range along X-axis was -50cm just above +40cm.All above mentioned distances are with respect to camera coordinates system,which is at (0,0,0) in both figures. So our camera calibration volume was approximately 90cmX70cmX200cm.

Camera calibration parameters estimated through experiments for our **Logitech Quickcam sphere AF** web- cam are tabulated:

Parameter	**Estimated Value**
f_x	1.3621520655751813e+03
f_y	1.3721887437041617e+03
c_x	8.0388406130158353e+02
c_y	5.9006614302109688e+02
k_1	7.2968012425660977e-02
k_2	-1.4332208217854106e-01
k_3	0.0
p_1	0.0
p_2	0.0
Reprojection error*	0.195039

Here,

$f_x : focal\ length\ of\ camera\ lens\ (expressed\ \text{in}\ number\ of\ pixels\ \ along\ X\text{-}axis)$

$f_y : focal\ length\ of\ camera\ lens\ (expressed\ \text{in}\ number\ of\ pixels\ \ along\ Y\text{-}axis)$

$(c_x, c_y) : Pixel\ coordinates\ of\ principal\ point$

$(k_1, k_2, k_3) : Radial\ distortion\ coefficients$

$(p_1, p_2) : Tangential\ distortion\ coefficients$

***Reprojection error**:As a measure of accuracy of fitted camera parameters,re-projection error is used. This is a geometric error,which quantitatively represents the euclidean distance between actual projection of a 3D point(the detected corners) and its projection computed using the estimated parameters.

2.3.Projector calibration:

As already mentioned in Section-1.2,projector calibration problem can be regarded as an *inverse-camera* calibration problem. Consequently,same procedure as of camera calibration has been used to calibrate the projector. Basically the idea is to use camera as a feedback device which will provide the 3D coordinates for the 2D features projected by the projector,hence providing the 3D-2D point correspondence between world-coordinate system & projector coordinate system needed for projector calibration similar to camera calibration process.

2.3.1.Procedure:

1.Camera-to-world homography is is computed once.
2.For each view,
a).Projector is physically displaced allowing for rotations & translation with respect to world coordinate system and a checkerboard pattern is projected.
b).Camera captures the projected features and detects their coordinates.
c).Using camera-to-world homography,corresponding 3D coordinates are computed.
d).These 3D coordinates along with the 2D coordinates of projected features(in projector buffer) forms the required 3D-2D correspondence for this view.
3.After sufficient number of such views(typically greater than 10) are captured projector calibration is performed using same algorithm as for camera calibration.

Figure-15 shows few views used for projector calibration,projected checkerboard pattern has board dimension 10X8 and has each square of size of 75 pixels X 75 pixels:

2.3.2.Results & visualization:

Figure-16 shows the snapshot of views used for projector calibration:

(a)

(b)

Here,figure 14(a) shows that calibration range for along X-axis is -100cm to +150cm along Z-axis it is -50cm to +250cm.Figure 14(b) shows calibration range along Y-axis to be from -125cm to +225cm.All measurements are with respect to camera-coordinate system which is shown at (0,0,0) in the figures. Hence calibration volume for projector was 250cmX400cmX300cm.

Calibration parameters for **Sharp PG F200X** projector computed by performing above steps are tabulated:

Parameter	**Estimated Value**
f_x	2261.709884536328
f_y	2262.798800587533
c_x	522.6660252563131
c_y	713.8403274734408
k_1	0.0
k_2	0.0
k_3	0.0
p_1	0.0
p_2	0.0

Explanation of calibration parameters has already been given in section-2.2.2.

2.4:Projector-camera relative calibration:

As mentioned in an end note in section:1.2.3,relative calibration of projector with respect to camera is needed to bring both projector and camera to a common coordinate system which is a implicit assumption in triangulation. In this work we used extrinsic calibration function available in OpenCV to determine the extrinsic parameters(i.e., rotation translation of projector coordinate system with respect to camera coordinate system).Rigid body transformations of camera & projector coordinate systems with respect to world coordinate system allow for determining the transformations between them(refer Note on relative calibration in section 1.2.3 for underlying equations)

2.4.1.Procedure

1.Extrinsic calibration of camera with respect to world coordinate system:

a).A known reference plane(whose dimensions are known) containing world coordinate system is captured & corners are detected.

b).These detected features are used for computing camera-to-world homography.

c).Using the 3D-2D correspondences determined above,extrinsic calibration of camera is performed.

2.Extrinsic calibration of projector with respect to world coordinate system:

a).Projector projects a checkerboard pattern,which is captured by the camera.

b).Using the camera-to-world homography computed above,3D coordinates for the projected feature points can be computed.

c).These 3D-2D correspondences are used for computing relative rotation & translation of projector coordinate system with respect to world coordinate system.

3.Relative rotation & translation of projector coordinate system with respect to camera coordinate system is computed(refer Note on relative calibration in section 1.2.3 for underlying equations)

2.4.2.Result & Visualization of extrinsic calibration & camera-projector relative calibration:

Estimated extrinsic calibration parameters for a sample camera-projector geometry are tabulated below:

Camera extrinsic parameters:

Parameter	**Value**
r_1^{wc}	-1.2784210332916737e-01
r_2^{wc}	2.3912072267449411e-01
r_3^{wc}	3.4797656010762248e-02
t_x^{wc}	-9.2073852583481778e+02
t_y^{wc}	-8.3457789450135158e+02
t_z^{wc}	2.9605565044207369e+03

Projector extrinsic parameters:

Parameter	**Value**
r_1^{wp}	-4.7036795791757673e-02
r_2^{wp}	-2.4555405743111489e-01

r_3^{wp}	7.4849590595966091e-03
t_x^{wp}	-1.0540252780850051e+03
t_y^{wp}	-1.2349689785957287e+03
t_z^{wp}	2.2810244571695575e+03

Rotation & translation transformation from projector to camera coordinate system:

Parameter	**Value**
r_1^{pc}	-0.085974
r_2^{pc}	0.484353
r_3^{pc}	0.005884
t_x^{pc}	-1.0802945093579142e+03
t_y^{pc}	1.8886527829014437e+02
t_z^{pc}	3.5934597865968681e+02

$$
\begin{aligned}
&\textit{Here},\\
&(r_1^{wc}, r_2^{wc}, r_3^{wc}): \textit{Rotation transformation} \text{ from } \textit{world coordinate system} \text{ to } \textit{camera coordinate system}\\
&(t_x^{wc}, t_y^{wc}, t_z^{wc}): \textit{Translation transformation} \text{ from } \textit{world coordinate system} \text{ to } \textit{camera coordinate system}\\
&(r_1^{\mathrm{wp}}, r_2^{\mathrm{wp}}, r_3^{\mathrm{wp}}): \textit{Rotation transformation} \text{ from } \textit{world coordinate system} \text{ to } \textit{projector coordinate system}\\
&(t_x^{\mathrm{wp}}, t_y^{\mathrm{wp}}, t_z^{\mathrm{wp}}): \textit{Translation transformation} \text{ from } \textit{world coordinate system} \text{ to } \textit{projector coordinate system}\\
&(r_1^{pc}, r_2^{pc}, r_3^{pc}): \textit{Rotation transformation} \text{ from } \textit{projector coordinate system} \text{ to } \textit{camera coordinate system}\\
&(t_x^{pc}, t_y^{pc}, t_z^{pc}): \textit{Translation transformation} \text{ from } \textit{projector coordinate system} \text{ to } \textit{camera coordinates system}\\
&\textit{Rotation vectors are expressed} \text{ in } \textit{radians} \text{ in } \textit{axis-angle form}, \textit{translation expressed} \text{ in } \textit{'mm'}
\end{aligned}
$$

View used for extrinsic calibration of projector camera system is depicted in **Figure-17:**

Figure-17(a) shows the actual projector-camera-world configuration:

<u>Figure-17(b)</u> shows the plot(plan-view) of estimated projector-camera-world configuration:

Here 'green' object is the top-view of projected checkerboard. Estimated positions of world-coordinate system,projector-coordinate system and camera-coordinate system are also shown.

2.5.1.Experiment procedure:

In this section we define the procedures used for experiments. Following sections report their results and our conclusions. It should be admitted that defining true values is a fundamentally challenging task in most cases and depends upon level of accuracy desired and should be decided based on multiple measurements.

Test procedure for camera calibration:

1.Perform Camera calibration 5 times(if time permits we would like to attempt for more number of times)

2.Simple test feature-points shown in **Figure-18** were marked on a board. Coordinates of these features were physically measured using Engineering drawing Drafter with respect to a world coordinate system and then it was reprojected to camera image.

3.Difference between true coordinates(observed coordinates of feature markers in captured image) and estimated coordinates was measured.

Figure-18

Test procedure for projector calibration:
For projector calibration,we initially used OpenCV algorithm but we were not getting repeatable values for parameters,hence decided to compare OpenCV with a tool available in public domain VPCLib**[46]** which is developed only for video projector calibration. We have observed repeatable results using that tool,which gives indication regarding its numerical stability. Test procedure used is as follows:
1.Perform projector calibration 5 times using both VPCLib & OpenCV.
2.Select candidate calibration value sets from both OpenCV & VPCLib. Criteria for selection was *minimum reprojection error* for OpenCV and *minimum average error* for VPCLib.
3.For both calibration value sets perform following steps:
3.1.1.Project a white image undistorted by calibration values from that particular method(OpenCV or VPCLib).
3.1.2.Measure world coordinates of end-points of projection region.
3.1.3.Project the measured 3D coordinates to projector image space using calibration matrices from particular method.
3.1.4.Compute difference between actual image coordinates and projected coordinates.

2.5.2.Results & conclusion:

In this section we report the results of experiments performed for evaluating accuracy of camera & projector calibration. Results are organized as follows:
2.5.2.1.Camera calibration values using OpenCV method
2.5.2.2.Measured re projection errors for all camera calibration values
2.5.2.3.Projector calibration values using OpenCV method
2.5.2.4.Projector calibration values using VPCLib method
2.5.2.5.Measured re projection errors for selected OpenCV & VPCLib calibration values

2.5.2.1.Camera calibration values using OpenCV method

Calibration index:1

Parameter	**Estimated Value**
f_x	1.3621520655751813e+03
f_y	1.3721887437041617e+03
c_x	8.0388406130158353e+02
c_y	5.9006614302109688e+02
k_1	7.2968012425660977e-02

k_2	-1.4332208217854106e-01
k_3	0.0
p_1	0.0
p_2	0.0
Reprojection error	0.195039

Calibration index:2

Parameter	**Estimated Value**
f_x	1.3394396280192989e+03
f_y	1.3539199363934806e+03
c_x	8.0292494976307501e+02
c_y	5.7721082825793462e+02
k_1	8.8366180798732805e-02
k_2	-1.7310342998076195e-01
k_3	0.0
p_1	0.0
p_2	0.0
Reprojection error	0.532193

Calibration index:3

Parameter	**Estimated Value**
f_x	1.3428715282729361e+03
f_y	1.3584222473454333e+03
c_x	7.8957563219940960e+02
c_y	5.7848356358501474e+02
k_1	8.7694140797461184e-02
k_2	-1.5996723053402206e-01
k_3	0.0
p_1	0.0
p_2	0.0
Reprojection error	0.608018

Calibration index:4

Parameter	**Estimated Value**
f_x	1.3765627464585434e+03
f_y	1.3922684326499134e+03
c_x	8.0150514505119986e+02
c_y	5.9057797590413520e+02
k_1	8.4463975772120184e-02
k_2	-1.3711673622718182e-01
k_3	0.0
p_1	0.0
p_2	0.0
Reprojection error	0.631338

Calibration index:5

Parameter	**Estimated Value**
f_x	1.3791179422452112e+03
f_y	1.3938279801969302e+03
c_x	7.9267040069006441e+02
c_y	5.8264512657959381e+02
k_1	9.9947295085227053e-02
k_2	-1.7451991268312619e-01
k_3	0.0
p_1	0.0
p_2	0.0
Reprojection error	0.631149

2.5.2.2.Measured reprojection errors for all camera calibration values:

Feature number	X_t	Y_t	X_e	Y_e

Here,

X_t :True X-coordinate in image

Y_t :True Y-coordinate in image

X_e :Estimated X-coordinate in image

Y_e :Estimated Y-coordinate in image

2.5.2.3.Projector calibration values using OpenCV method:

Calibration index:1

Parameter	Estimated Value
f_x	1.9369380486958357e+03
f_y	1.9381828066681851e+03
c_x	4.5746510194973888e+02
c_y	8.3596819835931569e+02
k_1	-1.0720876478244606e-02
k_2	1.2615178219933475e-02
k_3	0
p_1	0
p_2	0
Reprojection error*	0.475780

Calibration index:2

Parameter	Estimated Value
f_x	1.7814006649487510e+0
f_y	1.7918789482940488e+03
c_x	4.0173057112032899e+02
c_y	8.8841624255848751e+02
k_1	-1.1500251469408793e-02
k_2	1.4210412739543805e-02
k_3	0
p_1	0
p_2	0
Reprojection error*	0.581966

Calibration index:3

Parameter	Estimated Value
f_x	2.0831956452706008e+03
f_y	2.0854121047637082e+03
c_x	4.6363756337528497e+02
c_y	8.0820825154043143e+02
k_1	-1.7213160944764473e-02

k_2	1.1074476578118197e-02
k_3	0
p_1	0
p_2	0
Reprojection error*	0.538837

Calibration index:4

Parameter	**Estimated Value**
f_x	2.1915975297060718e+03
f_y	2.1901581660324164e+03
c_x	5.2006879655063312e+02
c_y	7.6523623648407715e+02
k_1	-3.2240817958796610e-02
k_2	1.0740659347209615e-01
k_3	0
p_1	0
p_2	0
Reprojection error*	0.381278

Calibration index:5

Parameter	**Estimated Value**
f_x	1.7638771051966505e+03
f_y	1.7732382589160559e+03
c_x	4.5657558285762627e+02
c_y	8.6367156511794008e+02
k_1	-1.2006183778191831e-02
k_2	1.2381413265324584e-02
k_3	0
p_1	0
p_2	0
Reprojection error*	0.393569

2.5.2.4.Projector calibration values using VPCLib method

Calibration index:1

Parameter	**Estimated Value**
f_x	2235.394222873481
f_y	2244.855515860762
c_x	531.5207202540331
c_y	726.8468421755762
k_1	0
k_2	0
k_3	0
p_1	0
p_2	0
Reprojection error	0.1896432588713265

Calibration index:2

Parameter	**Estimated Value**
f_x	2199.903315511075
f_y	2200.138617081568
c_x	530.5907629882773
c_y	739.1532412538903
k_1	0
k_2	0
k_3	0
p_1	0
p_2	0
Reprojection error	0.1856249020408495

Calibration index:3

Parameter	**Estimated Value**
f_x	2252.57153193369
f_y	2244.085752505491
c_x	536.1616011952278
c_y	733.9458602735594
k_1	0
k_2	0
k_3	0

p_1	0
p_2	0
Reprojection error	0.1917789987630331

Calibration index:4

Parameter	**Estimated Value**
f_x	2172.638904896803
f_y	2175.236629053769
c_x	521.0496190520309
c_y	743.0313176389982
k_1	0
k_2	0
k_3	0
p_1	0
p_2	0
Reprojection error	0.183725652779798

Calibration index:5

Parameter	**Estimated Value**
f_x	2088.28075253723
f_y	2097.154392652829
c_x	505.2099459523724
c_y	779.0456373293866
k_1	0
k_2	0
k_3	0
p_1	0
p_2	0
Reprojection error	0.181894743306481

2.5.2.5.Measured reprojection errors for selected OpenCV & VPCLib calibration values:

OpenCV method:

Vertex number	X_t	Y_t	X_e	Y_e
1	0	0	-1.79459	2.595
2	1023	0	1034.96	-3.23931
3	0	767	-1.75	7.729396
4	1023	767	1042.627	7.7024

VPCLib method:

Vertex number	X_t	Y_t	X_e	Y_e
1	0	0	-6.2822	2.539
2	1023	0	1030.98	-3.2279
3	0	767	-8.5061	775.074
4	1023	767	1040.66	771.5953

Here,

X_t :True X-coordinate in image

Y_t :True Y-coordinate in image

X_e :Estimated X-coordinate in image

Y_e :Estimated Y-coordinate in image

From the above values we observed that calibration values from OpenCV method were more close to the true values hence for further computations we decided to use OpenCV parameters.

CHAPTER 3: Correspondence

Structure of chapter:

3.1.Correspondence problem:Overview

3.2.Procedure
3.2.1.Pattern generation
3.2.2.Pattern projection & capture
3.2.3.Wrapped phase computation
3.2.4.Unwrapped phase computation
3.2.5.Absolute phase computation(recovery of camera-projector pixel-to-pixel correspondence)

3.3.Experiment

3.4.Results

3.1.Correspondence problem:

As mentioned in **Chapter-1**,correspondence problem is concerned with recovery of pixel-to-pixel correspondence between camera & projector which means to know about pair of projector & camera pixels which will see same point in the 3D space. In this work,correspondence problem was approached through a hybrid technique which combines both phase-shift & binary coded patterns. It gives higher spatial resolution of phase-shift technique and high noise robustness & considerable reduction in correspondence computation complexity of binary-coded patterns. Implementation details of this approach will be described in following sections.

3.2.Description:

In this work we have developed modules for performing following functions:

3.2.1.Pattern generation:

This module allows for generation of sinusoidal interference fringe patterns of arbitrary phase shift,fringe width & binary coded patterns of arbitrary number of codewords. Binary coded patterns are generated keeping width of a sinusoid period in mind because we want to resolve among individual sinusoid periods across a pattern using binary coded patterns,so bit-width of a binary coded pattern must be equal to the period of a sinusoid. In our implementation we have targeted to perform ray-ray triangulation(computing intersection of corresponding optical rays from projector & camera pixels) hence we need to project horizontal & vertical versions of both fringe & binary coded patterns. **Figure-19** shows horizontal & vertical fringe and binary coded patterns. Binary coded patterns were designed using following relations:

$$Number\ of\ codewords(vertical) = \frac{(Projector\ X\text{-}resolution)}{(period\ width\ for\ fringe)}$$

$$Number\ of\ codewords(horizontal) = \frac{(Projector\ Y\text{-}resolution)}{(period\ width\ for\ fringe)}$$

$$Number\ of\ vertical\ patterns = \frac{(\log(Number\ of\ codewords(vertical)))}{(\log(2))}$$

$$Number\ of\ horizontal\ patterns = \frac{(\log(Number\ of\ codewords(horizontal)))}{(\log(2))}$$

Figure-19:

Vertical fringe patterns:

Horizontal fringe pattens:

Vertical binary-coded patterns:

Horizontal binary-coded patterns:

[redacted]

[redacted]

3.2.2.Pattern projection & capture:

This module is developed for projection of binary coded & fringe patterns and to capture the patterns in sequential order. Currently it has only interactive control to project & capture patterns,but future version will include a synchronizing function to determine a common frequency of pattern projection & capture. In addition this module allows to control brightness of projector & camera. It allows to manipulate shutter-speed of camera hence facilitating to adapt according to varying illumination conditions. It should be noted that all projected & captured images are undistorted(i.e. camera/projector lens distortion effects are reduced) using the projector & camera intrinsic calibration parameters respectively.

Figure-20 shows the captured binary & fringe patterns:

Vertical fringe & binary-coded patterns:

Horizontal phase-shifted & binary-coded patterns:

3.2.3.Phase wrapping:

It should be noted that projected sinusoidal fringe pattern is periodic in nature. It implies that there exists ambiguity in determining the phase of any point across the pattern. This ambiguity is manifested in the illumination model used in Phase-shift interferometry technique. Following equations show the assumed illumination model at every camera pixel for three phase-shift technique**[30]** for both horizontal & vertical fringe patterns:

$$I_1 = I_{dc} + I_{mod} * \cos(\Phi - \Psi)$$
$$I_2 = I_{dc} + I_{mod} * \cos(\Phi)$$
$$I_3 = I_{dc} + I_{mod} * \cos(\Phi + \Psi)$$

where ,

I_i : *Captured intensity when* i^{th} *pattern is projected*

Φ : *Wrapped phase*

I_{dc} : *Background intensity*

I_{mod} : *Intensity modulation*

Ψ : *Constant phase shift between successive patterns*

Using this system of equations Φ can be estimated at every pixel assuming $\Psi = 2\frac{\Pi}{3}$:

$$\Phi = \tan^{-1}\left(\sqrt{3}\frac{(I_1 - I_3)}{(2I_2 - I_1 - I_3)}\right),$$

where , $-\Pi < \Phi < \Pi$

Since Φ lies from $-\Pi$ to $+\Pi$,information regarding actual phase at every point is mapped to this range hence it is called *wrapped* phase. It cannot be resolved for correct corresponding projector pixel unless we compute the corresponding period number,a task performed by module described in **Section-3.4.4.Figure-21** shows the wrapped phase computed from the captured phase shifted fringe images:

Vertical wrapped phase

Horizontal wrapped phase

3.2.4.Pixel decoding & phase unwrapping:

As mentioned in **section 3.4.3**,periodic nature of sinusoidal interference fringes,creates ambiguous condition in which multiple pixels in camera image can have same wrapped phase there is need to assign distinct period numbers to such candidates thereby eliminating the ambiguity. This step is called phase unwrapping. We developed a module to perform phase unwrapping which uses information from binary code projected onto the object to recover this period number. It should be noted here that wrapped & hence unwrapped phase is computed in both vertical & horizontal directions. Hence after these is a pair of unwrapped phase associated with each valid camera pixel. Here a camera pixel is called '*valid*' iff it corresponds to the region on which pattern is projected and has a Signal/Noise ratio greater than a threshold(this threshold is environment illumination dependent in current implementation).Here unwrapped phase is calculated using information from both fringe projection & binary coded techniques as:

$$\phi(x,y)=\Phi(x,y)+2\Pi*C(x,y)$$

$$where,$$

$$\phi(x,y):Unwrapped\ phase\ at\ pixel(x,y)$$

$$C(x,y):Decoded\ binary\ code\ at\ pixel(x,y)$$

This equation is applied for both horizontal & vertical phase unwrapping. **Figure-22** represents the unwrapped phase corresponding to the wrapped phase shown in Figure-21:

Vertical unwrappped phase

Horizontal unwrapped phase

3.2.5.Correspondence computation using absolute phase:

Once the unwrapped phase is computed,camera-projector pixel-pixel correspondences can be computed directly by searching for pixels in projector with same vertical & horizontal unwrapped phase values as that of corresponding *valid* camera pixel. It is computed as

3.3.Experiment:

This experiment is aimed at determining the accuracy of stereo-correspondence computed using the structured light technique implemented in this work. We use the calibration parameters selected after experiment described in chapter-2.In this work,stereo correspondence is computed using Phase-shift technique which also brings multiple issues related to its acquisition affecting accuracy of 3D reconstruction. One major issue in fringe-projection technique is non-linear gamma of projector-camera system. Without correction of gamma in projector,projection of sinusoidal pattern adds higher-order harmonics into the projected signal as reported by [**46**],**[30]** & many others. Its effect is immediately visible in the 3D reconstruction. **Figure-23** shows a snapshot of 3D reconstruction which clearly exhibits a *wavy-nature* in the output although scanned object was a plane. Although there are other sources of error also as described by **[49]** in Phase-shift interferometry like:quantization error in both projector & camera,stray reflections,intensity fluctuations error,frequency stability error,error due to projector & camera vibrations. But in this work we have limited the scope to work on gamma-calibration problem only. Future work will address other issues as well.

To keep the experiment simple and less prone to error due to issues not under consideration we decided to manually vary the projector gamma and arrive at an optimal value instead of developing an algorithm which may add its own artifacts which may deviate the focus from the parameter under consideration(i.e.,gamma value).It should be noted that the gamma calculated here is for camera-projector system and not for projector alone.Figure-23(a): Shows a plane which was initially reconstrucuted without doing any gamma correction:

Figure-23(b) shows a zoomed in view of 3D reconstruction of a plane shown in figure-23(a) exhibiting a *wavy*-pattern:

Procedure:

1.For a value of gamma:

a).Project the pattern

b).Capture and compute wrapped & unwrapped phase.

c).Compute stereo-correspondence from unwrapped phase.

d).Map the unwrapped phase to projector image using camera-projector correspondence.

e).Compute the ideal unwrapped phase at each projector pixel.

f).Compute the deviation of mapped unwrapped phase with respect to ideal unwrapped phase using Root-Mean-Absolute**[46]** measure of error for these correspondence deviations. The value of gamma for which Root-Mean-Absolute error is minimum is regarded as most accurate gamma value for projector camera system.

Here we selected *Root-Mean-Absolute error*(RMAE) rather than the frequently used *Root-Mean-Square error*(RMSE) because it is a function of average error whereas RMSE is function of 3 characteristics of error**[47]:**distribution of error magnitude,average error magnitude and square root of number of measurements.

Figure-24 shows projected fringes for various values of gamma:

Gamma=0.588 Gamma=1.0 Gamma=1.588

Gamma=2.041

Gamma=2.559

Gamma=3.012

3.4.Results:

Following table shows the result of applying various gamma values in terms of Root-Mean-Absolute error for vertical and horizontal unwrapped phase:

Gamma value	RMAE(Vertical unwrapped phase)	RMAE(Horizontal unwrapped phase)

1.0	0.050124	0.050324
1.588	0.049797	0.049764
2.041	0.049079	0.049082
2.171	0.049142	0.049143
2.365	0.056596	0.049097
2.559	0.049123	0.049044
2.624	0.049128	0.049174
2.882	0.049223	0.049220
3.012	0.049209	0.049215
3.335	0.056663	0.056808
3.529	0.049347	0.049223
3.918	0.049404	0.049263

After performing above experiment 3 times we observed that there is a dip in RMAE value between **Gamma=1.0 to Gamma=2.041** after which there is again a rise in the RMAE value. Further tests indicated that optimal Gamma value for the system lies somewhere between **2.041-2.559.** But we also observed that for **Gamma=2.041**

RMAE measure is lowest but corresponding 3D reconstruction still show wavy-appearance although when **Gamma=2.559** it is apparently removed. It implied that either Gamma is not the sole reason for such artifact in 3D point cloud or the RMAE criteria is not suitable for this experiment. We evaluated same experiment data with Root-Mean-Square error mesure also but with same conclusion.**Figure-24** shows the reconstructed plane after setting gamma=2.559 which shows improvement in reconstruction as compared to figure 23(b):

Later experiments have shown that even for the optimal gamma value this problem is not solved for all surfaces especially for very bright surfaces which seems to be due to pixel saturation in camera,as we observed that problem to be solved under higher camera shutter-speeds(i.e.,lower exposure period per frame).

CHAPTER 4: Triangulation

Structure of chapter:

4.1.Triangulation:Overview
4.2.Procedure
4.3.Visualization of Point-clouds
4.4.Experiments
4.5.Results

4.1.Triangulation:Overview

For triangulation we developed module for solving system of linear equations described in triangulation section in chapter-2 as *Linear-LS*. In our setup since we kept the world coordinate system origin at a reference plane board corner,all measurements(3D coordinates) are with respect to it. It should be noted that,position of world coordinate system is not disturbed after relative extrinsic calibration of camera & projector system. Here we rephrase the equations**[48]** relevant for 3D reconstruction:

Let (u_i^c, v_i^i) and (u_i^p, v_i^p) be the corresponding camera and projector pixels which will see (x_i^w, y_i^w, z_i^i) in world. Let A_c, A_p be the Projective transformations from world-coordinate system to camera & projector coordinate system respectively. In homogeneous coordinate representation following equations result:

For camera:

$$\begin{bmatrix} w_c * u_i^c \\ w_c * v_i^c \\ w_c \end{bmatrix} = \begin{bmatrix} a_{1,1}^c & a_{1,2}^c & a_{1,3}^c & a_{1,4}^c \\ a_{2,1}^c & a_{2,2}^c & a_{2,3}^c & a_{2,4}^c \\ a_{3,1}^c & a_{3,2}^c & a_{3,3}^c & 1 \end{bmatrix} \begin{bmatrix} x_i^w \\ y_i^w \\ z_i^w \\ 1 \end{bmatrix}$$

Similarly,for projector:

$$\begin{bmatrix} w_p * u_i^p \\ w_p * v_i^p \\ w_p \end{bmatrix} = \begin{bmatrix} a_{1,1}^p & a_{1,2}^p & a_{1,3}^p & a_{1,4}^p \\ a_{2,1}^p & a_{2,2}^p & a_{2,3}^p & a_{2,4}^p \\ a_{3,1}^p & a_{3,2}^p & a_{3,3}^p & 1 \end{bmatrix} \begin{bmatrix} x_i^w \\ y_i^w \\ z_i^w \\ 1 \end{bmatrix}$$

In form of equations we have:

$$\text{For camera},$$
$$w_c * u_i^c = a_{1,1}^c * x_i^w + a_{1,2}^c * y_i^w + a_{1,3}^c * z_i^w + a_{1,4}^c$$
$$w_c * v_i^c = a_{2,1}^c * x_i^w + a_{2,2}^c * y_i^w + a_{2,3}^c * z_i^w + a_{2,4}^c$$
$$w_c = a_{3,1}^c * x_i^w + a_{3,2}^c * y_i^w + a_{3,3}^c * z_i^w + 1$$
$$\text{For projector},$$
$$w_p * u_i^p = a_{1,1}^p * x_i^w + a_{1,2}^p * y_i^w + a_{1,3}^p * z_i^w + a_{1,4}^p$$
$$w_p * v_i^p = a_{2,1}^p * x_i^w + a_{2,2}^p * y_i^w + a_{2,3}^p * z_i^w + a_{2,4}^p$$
$$w_p = a_{3,1}^p * x_i^w + a_{3,2}^p * y_i^w + a_{3,3}^p * z_i^w + 1$$

$$\text{Eliminating } w_p \text{ and } w_c \text{ we get},$$
$$\text{For camera},$$
$$u_i^c - a_{1,4}^c = (a_{1,1}^c - u_i^c * a_{3,1} * c) x_i^w + (a_{1,2}^c - u_i^c * a_{3,2}^c) * y_i^w + (a_{1,3}^c - u_i^c * a_{3,3}^c) * z_i^w$$
$$v_i^c - a_{2,4}^c = (a_{2,1}^c - v_i^c * a_{3,1}^c) * x_i^w + (a_{2,2}^c - v_i^c * a_{3,2}^c) y_i^w + (a_{2,3}^c - v_i^c * a_{3,3}^c) * z_i^w$$
$$\text{For projector},$$
$$u_i^p - a_{1,4}^p = (a_{1,1}^p - u_i^p * a_{3,1} * p) x_i^w + (a_{1,2}^p - u_i^p * a_{3,2}^p) * y_i^w + (a_{1,3}^p - u_i^p * a_{3,3}^p) * z_i^w$$
$$v_i^p - a_{2,4}^p = (a_{2,1}^p - v_i^p * a_{3,1}^p) * x_i^w + (a_{2,2}^p - v_i^p * a_{3,2}^p) y_i^w + (a_{2,3}^p - v_i^p * a_{3,3}^p) * z_i^w$$

$$\text{Rewriting equations in matrix form } PV = F \text{ where},$$
$$P = \begin{bmatrix} (a_{1,1}^c - u_i^c * a_{3,1}^c) & (a_{1,2}^c - u_i^c * a_{3,2}^c) & (a_{1,3}^c - u_i^c * a_{3,3}^c) \\ (a_{2,1}^c - v_i^c * a_{3,1}^c) & (a_{2,2}^c - v_i^c * a_{3,2}^c) & (a_{2,3}^c - v_i^c * a_{3,3}^c) \\ (a_{1,1}^p - u_i^p * a_{3,1}^p) & (a_{1,2}^p - u_i^p * a_{3,2}^p) & (a_{1,3}^p - u_i^p * a_{3,3}^p) \\ (a_{2,1}^p - v_i^p * a_{3,1}^p) & (a_{2,2}^p - v_i^p * a_{3,2}^p) & (a_{2,3}^p - v_i^p * a_{3,3}^p) \end{bmatrix},$$
$$V = \begin{bmatrix} x_i^w \\ y_i^w \\ z_i^w \end{bmatrix},$$
$$F = \begin{bmatrix} (a_{3,4}^c * u_i^c - a_{1,4}^c) \\ (a_{3,4}^c * v_i^c - a_{2,4}^c) \\ (a_{3,4}^p * u_i^p - a_{1,4}^p) \\ (a_{3,4}^p * v_i^p - a_{2,4}^p) \end{bmatrix}$$
$$\text{Hence at each pixel}, \; V = (P^T * P)^{-1} * P^T * F \text{ which represents the 3D coordinate cooresponding to } (u_i^c, v_i^c) \,\&\, (u_i^p, v_i^p) \text{ in camera \& projector respectively.}$$

4.2.Procedure:

Here correspondence computed from absolute phase computation step is used along with the camera & projector intrinsic & extrinsic calibration parameters is used to compute depth. Relevant equations are already described in section 4.1.

4.3.Visualization of Point clouds:

To visualize,filter,smooth-out,down-sample and triangulate(mesh-generation from 3D data-set) the results Point cloud library(PCL) was deployed. Specifically,*PCL visualizer* was used for point cloud visualization. *Statistical outlier detection* algorithm was used

for filtering point cloud data. *Voxel-based* down-sampling method was used for reducing the number of points on rendered point cloud so as to provide relatively faster rendering rate. *Moving least squares(MLS)* algorithm was used for smoothing the point cloud data. *Greedy triangulation* algorithm was used to generate mesh output.3D point-clouds were also visualized in the ingeniously developed visualization tool **ANUVI-1.0.2.**

FIGURE-25:Shows snapshots of some sample 3D reconstructions using our system:

4.4.Experiment procedure:

We attempted to measure accuracy and resolution of our system using following procedure:

1.Mark end-points of straight lines of various lengths on a reference board.(**Figure-26** shows the reference board used)

2.For each decided depth from world coordinate system origin:-

a).Scan the reference surface.

b).Measure the distance between all marked end-points in the point-cloud. Here we performed it by interactively selecting the end-points in the point-cloud multiple times to reduce our own observation error.

Using this test we determined accuracy of measurement for various distances from our world-coordinate system origin and by measuring line-segments of various lengths we attempted to determine resolution of our system.

Figure-27:

4.5.Results:

Here we report current measurement accuracy[1] of our system for various depths determined by performing the experiment described above:

Experiment ID	True value	Measured value	Distance from origin[2]

1.We performed 5 measurements for each line-segment for each depth to reduce our own observation error. Mean value of measurement is reported here.

2.Distance from world coordinate system origin is approximate,alignment of measurement plane with respect to world-coordinate system may not be *exactly* parallel.

Figure-28 shows the position of measurement board with respect to camera for various depths used for experiment:

CHAPTER 5:
Conclusion,observations & proposed future work:

0.Study relation between calibration volume & measurement volume.
1.Synchronization of camera & projector.
2.Improve region growing to allow for larger region scan without seg fault.
3.Remove mis-alignment between phase-shift pattern & binary coded pattern
4.Implement gray coded pattern to see if decoding error reduces.
5.Correct the exposure control of webcam.
6.Provision for phase-unwrapping algorithm.
7.Analyse & compare previous 2 implemented triangulation methods.

APPENDIX-I
Accuracy comparison with Kinect:

Structure of appendix:

A.1.1.Description:Kinect

A.1.2.Experiment

A.1.3.Results

A.1Description:Kinect

Kinect is a natural user interface developed by Microsoft for XBOX360 & Windows Pcs(Windows 7 onwards).In its typical application it allows speech & gesture recognition based control,without need of physically touching the consoles. It is composed of a depth camera,a RGB camera both of which can capture at 640X480 at 30Hz and an infrared projector. In addition to it also contains Multi-array MIC for speech based control. The depth camera is a 11-bit monochromatic sensor,which provides 2048-levels of sensitivity. Kinect sensor is reported to have practical depth range from 1.2m-3.5m when used with XBOX360.Sensor has angular field-of-view of 57 degrees horizontally and 43 degrees vertically. Motorized sensor can be tilted up to 27 degrees up or down. Horizontal field of sensor at minimum viewing distance of ~0.8 m is ~87cm,vertical field is ~63cm resulting in resolution just over 1.3mm per pixel.

BIBILIOGRAPHY:

1.Microsoft Photosynth:
photosynth.net

2.Projective geometry tutorial
http://www.cse.iitd.ernet.in/~suban/vision/tutorial/node1.html

3.@ARTICLE{Faugeras95stratificationof,
author = {Olivier Faugeras},
title = {Stratification of 3-D vision: projective, affine, and metric representations},
journal = {Journal of the Optical Society of America A},
year = {1995},
volume = {12},
pages = {46548--4}
}

4.Triangulation
Richard I. Hartley, Peter Sturm
Computer Vision and Image Understanding, Volume 68, Issue 2, November 1997, Pages 146-157

5.@MISC{Saxena_3-dreconstruction,
author = {Ashutosh Saxena and Min Sun and Andrew Y. Ng},
title = {3-D Reconstruction from Sparse Views using Monocular Vision},
year = {}
}

6.Coded Structured Light for 3D-photography:an Overview
Asla Medeiros e Sa,Esdras Soares de Medeiros Filho,Paulo Cezar Carvalho,Luiz Velho

7.Tutorial on Coded Light Projection techniques
J.Salvi,J Pages
http://eia.udg.es/~qsalvi/Tutorial_Coded_Light_Projection_Techniques_archivos/v3_document.html

8.@ARTICLE{Salvi04patterncodification,
author = {Joaquim Salvi and Jordi Pagès and Joan Batlle},
title = {Pattern Codification Strategies in Structured Light Systems},
journal = {PATTERN RECOGNITION},
year = {2004},
volume = {37},
pages = {827--849}
}

9.J. L. Posdamer and M. D. Altschuler, Surface measurement by space-encoded projected beam systems, Comput. Graph. Image Processing 18(1), 1-17 (1982).

10.S. Inokuchi, K. Sato, and F. Matsuda, Range-imaging for 3-D object recognition, in International Conference on Pattern Recognition (International Association for Pattern Recognition, 1984), pp. 806808.

11.D. Caspi, N. Kiryati, and J. Shamir, Range imaging with adaptive color structured light, IEEE Trans. Pattern Anal. Mach. Intell. 20(5), 470-480 (May 1998).

12.@ARTICLE{Horn97towardoptimal,
author = {Eli Horn and Nahum Kiryati},
title = {Toward Optimal Structured Light Patterns},
journal = {Image and Vision Computing},
year = {1997},
volume = {17},
pages = {87--97}
}

13.Two-Dimensional Phase Unwrapping: Theory, Algorithms, and Software(BOOK)
Dennis C.Ghiglia,Mark D.Pritt

14.F. J. MacWilliams and N. J. A. Sloane, Pseudorandom sequences and arrays, Proc. IEEE 64(12), 1715-1729 (1976).

15.J. Le Moigne and A. M. Waxman, Multi-resolution grid patterns for building range maps,in Vision -85, Applied Machine Vision Conference (ASME) (Society of Manufacturing Engineers, 1985), pp. 22-39.

16.A. Osman Ulusoy, F. Calakli, and G. Taubin, One-shot scanning using De Bruijn spaced grids, in 2009 IEEE 12th International Conference on Computer Vision Workshops (ICCV Workshops) (IEEE, 2009), pp. 1786-1792.

17.B. Carrihill and R. Hummel. Experiments with the intensity ratio depth sensor. In Computer Vision, Graphics and Image
Processing, volume 32, pages 337--358. Academic Press, 1985

18.G. Chazan and N. Kiryati. Pyramidal intensityratio depth sensor. Technical report 121, Center for Communication and Information Technologies, Department of Electrical Engineering, Technion, Haifa, Israel, October 1995.

19.D.C.D. Hung. 3d scene modelling by sinusoid encoded illumination. Image and Vision Computing, 11:251--256, 1993.

20.J. Tajima and M. Iwakawa. 3D data acquisition by rainbow range finder. In International Conference on Pattern Recognition, pages 309--313, 1990

21.T. Sato. Multispectral pattern projection range finder. In Proceedings of the Conference on ThreeDimensional Image Captuer and Applications II, volume 3640, pages 28--37, San Jose, California, January 1999. SPIE.

22.@MISC{Pagès03overviewof,
author = {Jordi Pagès and Joaquim Salvi and Rafael Garcia and Carles Matabosch},
title = {Overview of coded light projection techniques for automatic 3D profiling},
year = {2003}
}

23.Stripe boundary codes
for real-time structured-light range scanning of moving
objects. In The 8th IEEE International Conference on
Computer Vision, pages II: 359366, 2001.

24.Recent progresses on real-time 3D shape measurement using digital fringe projection techniques
Song Zhang
Optics and Lasers in Engineering, Volume 48, Issue 2, February 2010, Pages 149-158

25.Fringe projection technique:Whither we are?
SS Gorthi, P Rastogi - Optics and Lasers in Engineering, 2009

26.'open-light'-Google code
http://code.google.com/p/open-light/

27.'Strutured-light'-Goole code
http://code.google.com/p/structured-light/

28.Dennis C. Ghiglia and Louis A. Romero, "Minimum Lp-norm two-dimensional phase unwrapping," J. Opt. Soc. Am. A 13, 1999-2013 (1996)

29.Tsai, Roger Y. (1987) :A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-Shelf TV Cameras
and Lenses,IEEE Journal of Robotics and Automation, Vol. RA-3, No. 4,
August 1987, pp. 323-344.

30. High-resolution, Real-time 3-D Shape Measurement, Ph.D. dissertation, Stony Brook University, Stony Brook, NY, 2005.

31.Z. Zhang. A flexible new technique for camera calibration. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(11):1330-1334, 2000

32.R.I.Hartley,An algorithm for self calibration from several viewes.In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition,pages 908-912,IEEE Computer Society Press

33.Q.-T Luong and O. Faugeras.Self-calibration of a moving camera from point correspondences and fundamental matrices.The International Journal of Computer Vsion,22(3):261-289,1997

34.B Caprile and V.Torre.Using Vanishing Points for Camera Calibration.The

International Journal Of Computer Vision,4(2):127-140,Mar.1990

35.R.Hartley.Self-calibration from multiple views with a rotating camera.In Proceedings of 3rd ECCV,volume 800-801 of 'Lecture notes in Computer Science' pages 471-478,Stockholm,Sweden,May 1994.Springer-Verlag

36.Camera Calibration Toolbox For Matlab
www.vision.caltech.edu/bouguetj/calib_doc/

37.A Four-step Camera Calibration Procedure with Implicit Image Correction - Heikkila and Silven, CVPR97

38.Minh Vo, Zhaoyang Wang, Bing Pan, and Tongyan Pan, "Hyper-accurate flexible calibration technique for fringe-projection-based three-dimensional imaging," Opt. Express 20, 16926-16941 (2012)

39.Appendix:Projective geometry for machine vision
J Mundy, A Zisserman - Geometric invariances in computer vision,MIT Press ,1992

40.@Book{Hartley2004,
author = "Hartley, R.~I. and Zisserman, A.",
title = "Multiple View Geometry in Computer Vision",
edition = "Second",
year = "2004",
publisher = "Cambridge University Press, ISBN: 0521540518"
}

41.Triangulation
http://en.wikipedia.org/wiki/Triangulation

42.OpenCV documentation on Camera calibration
http://docs.opencv.org/modules/calib3d/doc/calib3d.html

43. Learning OpenCV
Computer Vision with the OpenCV Library
By Gary Bradski, Adrian Kaehler
Publisher: O'Reilly Media
Released: September 2008

44.Link to doumentation of cvFindExtrinsicCameraParams2
http://opencv.willowgarage.com/documentation/c/calib3d_camera_calibration_and_3d_reconstruction.html#findextrinsiccameraparams2

45.R. Hartley, R. Gupta, and T. Chang, Stereo from uncalibrated cameras,
in Proceedings, IEEE Conference on Computer Vision and Pattern Recognition, 1992, pp. 761764.

46.Mean Absolute article
en.wikipedia.org/wiki/Mean_absolute_error

47.Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance
Cort J. Willmott, Kenji Matsuura

Climate Research Vol.30:79-82(2005)

FIGURE SOURCES:

F.1. Jason Geng, "Structured-light 3D surface imaging: a tutorial," Adv. Opt. Photon. 3, 128-160 (2011)

F.2.De Bruijn sequnces
http://feed-back.be/nick/?page_id=310

F.3.@MISC{Pagès03overviewof,
author = {Jordi Pagès and Joaquim Salvi and Rafael Garcia and Carles Matabosch},
title = {Overview of coded light projection techniques for automatic 3D profiling},
year = {2003}
}

F.4.Tsai, Roger Y. (1987) :A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-Shelf TV Cameras and Lenses,IEEE Journal of Robotics and Automation, Vol. RA-3, No. 4, August 1987, pp. 323-344.