

# Distortion Correction for Non-Planar Deformable Projection Displays through Homography Shaping and Projected Image Warping

by

Onoise Gerald Kio

**A thesis submitted in partial fulfilment for the requirements for the degree of**

**Doctor of Philosophy**

**at the University of Central Lancashire**

**August 2016**

# Declaration

## **Concurrent registration for two or more academic awards**


I declare that while registered as a candidate for the research degree, I have not been a registered candidate or enrolled student for another award of the University or other academic or professional institution.

## **Material submitted for another award**

I declare that no material contained in the thesis has been used in any other submission for an academic award and it is solely my own work.

## **Collaboration**

The work presented in this thesis was carried out at the Applied Digital Signal and Image Processing (ADSIP) Research Centre, University of Central Lancashire. The work described in the thesis is entirely the candidate's own work.

Signed: 

Type of Award: Doctor of Philosophy

Department: School of Engineering

## Abstract

Video projectors have advanced from being tools for only delivering presentations on flat or planar surfaces to tools for delivering media content in such applications as augmented reality, simulated sports practice and invisible displays. With the use of non-planar surfaces for projection comes geometric and radiometric distortions. This work dwells on correcting geometric distortions occurring when images or video frames are projected onto static and deformable non-planar display surfaces.

The distortion-correction process involves (i) detecting feature points from the camera images and creating a desired shape of the undistorted view through a 2D homography, (ii) transforming the feature points on the camera images to control points on the projected images, (iii) calculating Radial Basis Function (RBF) warping coefficients from the control points, and warping the projected image to obtain an undistorted image of the projection on the projection surface.

Several novel aspects of this work have emerged and include (i) developing a theoretical framework that explains the cause of distortion and provides a general warping pattern to be applied to the projection, (ii) carrying out the distortion-correction process without the use of a distortion-measuring calibration image or structured light pattern, (iii) carrying out the distortion-correction process on a projection display that deforms with time with a single uncalibrated projector and uncalibrated camera, and (iv) performing an optimisation of the distortion-correction processes to operate in real-time.

The geometric distortion correction process designed in this work has been tested for both static projection systems in which the components remain fixed in position, and dynamic projection systems in which the positions of components or shape of the display change with time. The results of these tests show that the geometric distortion-correction technique developed in this work improves the observed image geometry by as much as 31% based on normalised correlation measure. The optimisation of the distortion-correction process resulted in a 98% improvement of its speed of operation thereby demonstrating the applicability of the proposed approach to real projection systems with deformable projection displays.

# Table of Contents

Declaration.....	i
Concurrent registration for two or more academic awards.....	i
Material submitted for another award.....	i
Collaboration.....	i
Abstract.....	ii
Table of Contents.....	iii
List of Figures and tables.....	viii
Acknowledgement.....	xviii
Abbreviations.....	xix
<b>CHAPTER ONE: Project Overview.....</b>	<b>1</b>
1.1 Aims and Objectives.....	1
1.2 Methodology.....	5
1.3 Novelty of the Distortion-correction Approach.....	7
1.4 Organisation of Thesis.....	9
<b>CHAPTER TWO: Literature Review.....</b>	<b>11</b>
2.1 Introduction.....	11
2.2 Image Geometric Distortion: Causes and Correction .....	11
2.3 Projector-Camera Systems.....	14
2.3.1 Projector-Camera System Calibration.....	14
2.3.2 Correction of Geometric Distortion for Projected Displays.....	15
2.3.3 Correction of Geometric Distortion for Projected Displays Developed in this Work.....	21
2.4 Image Warping Techniques.....	23
2.5 Image Similarity Measurement.....	26
2.6 Chapter Summary.....	27

<b>CHAPTER THREE: Theory of Image Processing Techniques.....</b>	<b>29</b>
3.1 Introduction.....	29
3.2 Image Feature Detection Techniques.....	29
3.2.1 Image Edge Detection.....	30
3.2.2 Image Corner Detection.....	36
3.2.3 Blob Detection.....	38
3.3 Projections on Planar Surfaces.....	40
3.3.1 2D Homography.....	40
3.3.2 2D Projection onto the 2D Sensor Plane of a Camera.....	42
3.3.3 2D Transformations between a Projector, a Planar Projection Surface and a Camera.....	43
3.4 Straight Line Geometry.....	45
3.4.1 The Equation of a Straight 2D Line.....	46
3.4.2 Estimating the Distortion of a Straight 2D Line.....	47
3.5 Image Warping Techniques.....	48
3.5.1 Translation.....	48
3.5.2 Procrustes Transformation.....	49
3.5.3 Affine Transformation.....	49
3.5.4 Perspective Transformation.....	50
3.6 Image Warping with Radial Basis Function Interpolation.....	51
3.7 Image Similarity Measurement.....	53
3.7.1 Normalised Cross-correlation Coefficient (NCC).....	53
3.7.2 Phase Correlation.....	54
3.8 Chapter Summary.....	55
 <b>CHAPTER FOUR: Geometric Distortion Correction for Static Projection Displays</b>	<b>56</b>
4.1 Introduction.....	56
4.2 Illustrating Geometric Distortion of a Straight Line Projected Onto Non-planar Surface.....	57
4.2.1 Pictorial Illustration.....	57

4.2.2 Algebraic Representation of a Plane and a Cylinder.....	58
4.3 Geometric Distortion Measurement of Image Lines	
and Geometric Distortion Correction using a Calibration Image.....	62
4.3.1 The Calibration Image and Measure of Distortion of its	
Grid Lines.....	62
4.3.2 Measure of Distortion/Straightness of Straight Lines	
in an Image.....	65
4.3.3 Measuring the Distortion of Projected Horizontal and Vertical Lines	
of the Calibration Image.....	67
4.3.4 The Geometric Distortion-correction Process.....	70
4.3.5 Experimental Considerations.....	74
4.4 Results of Distortion-correction Tests with the Calibration Image.....	75
4.4.1 General Distortion-correction Results.....	75
4.4.2 Results of Tests to Determine the Number of Feature Points	
to Use in the Calibration Image.....	83
4.4.3 Results of Tests to Determine the Type of RBF to Use for	
Distortion Correction.....	87
4.4.4 Results of Distortion-correction Tests for Other Non-planar	
Surfaces, eg Two Planar Surfaces Meeting at Right Angles.....	88
4.5 Automatic Geometric Distortion-correction Without Using a Calibration	
Image.....	92
4.5.1 Edges and Corners at the Periphery of the Projected and	
Captured Images used as Feature Points for Distortion Correction.	93
4.5.2 Blobs in the Projected and Captured Images Used as Feature Points	
for Distortion Correction.....	94
4.5.3 Linear Modelling of Distortion.....	95
4.6 Results of Tests of Automatic Geometric Distortion-correction	
Without Using a Calibration Image.....	98
4.6.1 Edges and Corners at the Periphery of the Projected and	
Captured Images Used as Feature Points for Distortion Correction.	98

4.6.2 Distortion-correction using Blob Features.....	100
4.6.3 Linear Modelling of Distortion.....	103
4.7 Chapter Summary.....	109
<b>CHAPTER FIVE: Geometric Distortion Correction for Deformable and Dynamic Projection Systems.....</b>	<b>112</b>
5.1 Introduction.....	112
5.2 Illustrating Geometric Distortion of a Warped Image Line Projected onto a Deformable or Dynamic Quadric Surface.....	113
5.3 Distortion Correction for Real Dynamic Projection Systems.....	114
5.4 Results of Tests Carried out for Distortion-correction Techniques on Real Dynamic Projection Systems.....	116
5.5 Code Profiling and Optimisation towards Real-time Operation.....	123
5.5.1 Creating a Region-of-Importance (ROI) for Edge and Corner Detection.....	125
5.5.2 Estimating Corners from Edge Information.....	128
5.5.3 Using linear Interpolation to Approximate the Warping of the Projected Image.....	130
5.5.4 Greyscale versus RGB Processing.....	131
5.5.5 Parallel Processing.....	131
5.6 Results of Code Profiling and Optimisation for Real-time Performance..	132
5.6.1 Code Profiling Results Obtained from Creating an ROI for Edge Detection and 4 ROIs for Corner Detection.....	132
5.6.2 Code Profiling Results Obtained from Estimating Four Corners from Detected Edges.....	133
5.6.3 Code Profiling Results Obtained from Using Linear Interpolation to Approximate the Warping of the Projected Image.....	136
5.7 Golf Simulation.....	140
5.8 Results from Golf Simulation.....	141
5.9 Chapter Summary.....	144

<b>CHAPTER SIX: Conclusion and Recommendations.....</b>	<b>146</b>
6.1 Conclusion.....	146
6.2 Original Contributions.....	148
6.3 Practical Considerations.....	149
6.3.1 Practical Applications.....	149
6.3.2 Image Processing Considerations.....	150
6.4 Recommendations.....	151
<b>APPENDIX.....</b>	<b>153</b>
A1 Detection and Arranging Detected Corner Points of the Distorted Projected Calibration Image.....	153
A2 Obtaining feature points along a distorted edge.....	157
A3 Graphical User Interface (GUI).....	159
<b>REFERENCES.....</b>	<b>161</b>



## List of Figures and Tables

<b>Figure 1:</b> The components of a projector-camera system consisting of multiple projectors, two cameras, a computer and projection surface/display.....	2
<b>Figure 2:</b> Block diagram of a projector-camera system showing its components and their functions.....	3
<b>Figure 3:</b> Optical ‘see-through’ camouflage system applied to the backseat of a vehicle.....	4
<b>Figure 4:</b> An <i>aboutGolf</i> ltd golf simulator showing its projector, loose screen, and tee area.....	5
<b>Figure 5:</b> Components of a geometric distortion-correction system and process	7
<b>Figure 6:</b> Generalised block diagram describing the distortion-correction process for projected displays.....	17
<b>Figure 7:</b> Various projection surfaces used to test distortion-correction techniques mentioned in literature.....	23
<b>Figure 8:</b> Rapid change of univariate function indicates an edge.....	31
<b>Figure 9:</b> Laplacian Convolution Kernels.....	33
<b>Figure 10:</b> Comparison of various edge detectors to test their suitability for detecting projected image features.....	35
<b>Figure 11:</b> An image corner.....	36

<b>Figure 12:</b> Comparison of the Harris and Stephens corner detector with Shi and Thomasi minimum eigenvalue corner detector in terms of detected corners and algorithm speed.....	38
<b>Figure 13:</b> Detecting common blob features between a projected image and the camera image of the projection.....	40
<b>Figure 14:</b> The projection of points from a plane surface to respective points in two image planes.....	41
<b>Figure 15:</b> Image $I_p$ projected from the Projector Plane $P$ to form an image $I_s$ on the projection surface.....	44
<b>Figure 16:</b> A straight 2D line $l$ described by the equation $v = mu + c$ .....	46
<b>Figure 17:</b> Translation of a region of pixels in an image.....	48
<b>Figure 18:</b> Procrustes transformation of an image.....	49
<b>Figure 19:</b> Affine transformation of an image.....	50
<b>Figure 20:</b> Projective-transformed image with.....	50
<b>Figure 21:</b> A projection system consisting of a projector, camera, projection surface and computer.....	57
<b>Figure 22:</b> Geometric illustration of the projection of a 2D image line onto a 3D or non-planar surface.....	59

<b>Figure 23:</b> Plots of the profile curve of a horizontal line projected onto a quadric (cylindrical) surface.....	61-62
<b>Figure 24:</b> Calibration image consisting of 14×14 feature points projected onto the non-planar surface to measure and correct projected image distortion	64
<b>Figure 25:</b> Camera capture of the calibration image projected onto the non-planar projection surface showing the detected corner points, horizontal and vertical lines formed by linking two extreme corner points, and the quadrilateral formed by linking the corner points at the four corners of the image.....	64
<b>Figure 26:</b> A plot of average deviation of member feature points from the horizontal and vertical lines shown in figure 25.....	65
<b>Figure 27:</b> 200 Randomly-generated image lines.....	66
<b>Figure 28:</b> Plot of average deviations for lines shown in figure 27.....	67
<b>Figure 29:</b> Histogram showing the distribution of average deviation values for the 200 randomly-generated lines in figure 27.....	67
<b>Figure 30:</b> Histograms of distributions of average deviations of points from the projected vertical lines of the calibration image and average deviations of points from 200 randomly-generated straight image lines.....	68

<b>Figure 31:</b> Histograms of distributions of average deviations of points from the projected horizontal lines of the calibration image and average deviations of points from 200 randomly-generated straight image lines.....	69
<b>Figure 32:</b> Histograms of distributions of average deviations of points from the projected horizontal lines and vertical lines of the calibration image and average deviations of points from 200 randomly-generated straight image lines.....	69
<b>Figure 33:</b> Illustration of the respective positions of the distortion-corrected line $l$ , a line $l'$ perpendicular to it, and distorted feature point $p_d$ opposite point $p_d'$ and distortion-corrected point $p_r$ .....	71
<b>Figure 34:</b> Image of a calibration image showing correctly detected feature points, missing and unwanted feature points.....	74
<b>Figure 35:</b> Geometric distortion correction of the projection of a calibration image consisting of 196 feature points.....	76-77
<b>Figure 36:</b> Geometric distortion correction of the projection of a natural image replacing the calibration image and using the warping coefficients obtained from the calibration image.....	78-79
<b>Figure 37:</b> Average deviation of feature points from their respective horizontal and vertical lines on the observed images of projection of the calibration image.....	80
<b>Figure 38:</b> Normalised Correlation Coefficients (NCC) of the observed distorted and distortion-corrected projections.....	81

<b>Figure 39:</b> Peak NCC values for distorted and distortion-corrected projections using 100, 144 and 196 feature points.....	82-83
<b>Figure 40:</b> Results to demonstrate the need for having a sufficiently high number of feature points for the calibration image to achieve better geometric distortion correction.....	84-85
<b>Figure 41:</b> Curve sections with varying number of feature points to be used as RBF control points to warp the curve sections into the straight horizontal lines.....	86
<b>Figure 42:</b> Comparison of three RBF types based on the NCC values of distortion-corrected images.....	87
<b>Figure 43:</b> Distortion-Correction Results for 2 planar surfaces meeting at right angles.....	89-92
<b>Figure 44:</b> Illustration of a warped projected image showing the initial edges, the final edges after warping and the initial and final points along each edge.	96
<b>Figure 45:</b> Illustration of the assumed linear relationships between coordinates of points in the unwarped and warped images.....	97
<b>Figure 46:</b> Results of auto geometric distortion correction by RBF warping using 28 control points from the top and bottom edges of the projected image.....	98-100
<b>Figure 47:</b> Results of auto geometric distortion correction by RBF warping using blob features obtained from the SIFT algorithm.....	101-103

<b>Figure 48:</b> Box and whisker plots for a single image showing the absolute difference between horizontal and vertical pixel coordinates obtained from using a calibration image and coordinates obtained from linear interpolation of coordinate displacements.....	105
<b>Figure 49:</b> Comparison between pre-warped projected images by RBF coefficients obtained from using a calibration image and RBF coefficients obtained from linear interpolation of deviation of control points on the periphery of the projected image.....	105-106
<b>Figure 50:</b> Observed geometrically distorted projection and distortion-corrected projections for methods involving the use of the 196-point calibration image and that obtained from using 52 known control point displacements and 144 linear interpolated control point displacements.....	107-108
<b>Figure 51:</b> Improvement of peak NCC matching values between the desired observed projection and the observed distorted projection and distortion-correction by linear interpolation of control point displacement for 10 different poses of the projector, camera and shapes of the projection surface.....	108
<b>Figure 52:</b> Curves showing the profiles of the warped and unwarped projected lines and their respective profiles on the quadric projection surface before and after the change of shape of the projection surface	113-114
<b>Figure 53:</b> Distortion-correction for dynamic projection systems.....	115

<b>Figure 54:</b> Results for successively correcting, distorting, and correcting the observed projection by warping the projected image from its previously-warped state.....	117-119
<b>Figure 55:</b> Graph showing peak NCC values for successively correcting, distorting, and correcting the observed projection by warping the projected image from its previously-warped state.....	120
<b>Figure 56:</b> Results for successively correcting, distorting, and correcting the observed projection by warping the projected image from its unwarped state.....	120-122
<b>Figure 57:</b> Graph of peak NCC values obtained from successively correcting, distorting, and correcting the observed projection by warping the projected image from its unwarped state.....	123
<b>Figure 58:</b> Illustrating the principle of developing ROIs for faster edge and corner detection.....	125
<b>Figure 59:</b> Observed consecutive frames of a projection on a surface hit by a projectile in an experiment to investigate the deviation of feature points from one frame to the next on a deforming projection surface.....	126
<b>Figure 60:</b> Observed displacement of edge pixels in the horizontal and vertical image directions after striking projection surface with a projectile	127
<b>Figure 61:</b> Two-dimensional variation of the four corners of observed projected video frames on a deforming surface.....	128
<b>Figure 62:</b> Illustration of how to estimate corners from edges.....	129

<b>Figure 63:</b> Pictorial illustrations to show the closeness between estimated corners and actual corners.....	134
<b>Figure 64:</b> Comparison between values of estimated and true corner points using cosines of angles and absolute magnitude differences between their respective vectors.....	135
<b>Figure 65:</b> Comparison between 3 methods used to warp the 640 x 480 pixel sized projected image.....	137-138
<b>Figure 66:</b> Geometric distortion correction process for simulated golf	140
<b>Figure 67:</b> Projected image of a golf course and distortion-corrected frames of the projection after projectile hits the projection surface....	141-143
<b>Figure 68:</b> Peak NCC values for distorted and distortion-corrected frames of the golf-simulation test.....	144
<b>Figure 69:</b> A chosen corner point and its four closest neighbours.....	154
<b>Figure 70:</b> Flow diagram of steps used in producing an ordered set of corner points from an unordered set of points.....	156
<b>Figure 71:</b> Camera view of the calibration image showing misarranged and correctly-arranged corners.....	156-157
<b>Figure 72:</b> Edge image of the projection showing red non-distorted points of the top and bottom edges.....	157



<b>Figure 73:</b> A typical top peripheral section of the camera image of the projection showing the distorted edge with the a distorted feature point, the desired non-distorted edge with a non-distorted feature point, the perpendicular line, and other detected feature points in the image.....	158
<b>Figure 74:</b> GUI created in Matlab.....	160
<b>Table 1:</b> Convolution kernels for gradient-based edge detectors.....	32
<b>Table 2:</b> Examples of Radial basis functions.....	53
<b>Table 3:</b> Comparing approximate minimum and maximum deviations in 2 image directions for horizontal and vertical lines of the projected calibration image.....	65
<b>Table 4:</b> Comparison of three RBF types based on the average computation time to warp a 640×480 pixel image.....	88
<b>Table 5:</b> Average time taken for critical image distortion-correction operations to complete on Matlab.....	124
<b>Table 6:</b> Comparison between estimated mean coordinates of corners obtained from the first frame of a video and their actual mean coordinates obtained from the whole video duration.....	130
<b>Table 7:</b> Processing time saving resulting from creating ROIs for edge and corner detection.....	132
<b>Table 8:</b> Processing time saving resulting from creating 2 ROIs for edge detection and estimating corners from detected edges.....	133

**Table 9:** Processing time saving resulting from using linear interpolation to warp most of the projected image of size 640 x 480 pixels rather than evaluating the warping function to warp the whole image..... 136

**Table 10:** Summary table showing all image processing time savings obtained from code profiling for real-time performance..... 139

## **Acknowledgement**

I wish to thank God for giving me the strength, patience and enablement to carry out this work.

Special thanks to my darling wife Jumoke and daughter Annabelle, my mother Agatha, and members of my extended family for their support and encouragement.

My profound gratitude goes to the Niger Delta Development Commission of the Federal Government of Nigeria (NDDC) that sponsored my studies at the University of Central Lancashire, United Kingdom.

I wish to acknowledge the invaluable guidance of members of my supervisory team including my Director of Studies Professor Lik-Kwan Shark, and my co-supervisors Dr. Geoff Hall and Professor Bogdan Matuszewski.

# Abbreviations

**ASIFT:** Affine Scale-Invariant Feature Transform

**CFD:** Computational Fluid Dynamics

**CMM:** Coordinate Measuring Machine

**CSS:** Curvature Scale Space

**CT:** Computed Tomography

**DIA:** Document Image Analysis

**DoG:** Difference of Gaussians

**DirectAR:** Direct Augmented Reality

**FAST:** Features From Accelerated Segment Test

**GPU:** Graphics Processing Unit

**GUI:** Graphical User Interface

**GSIFT:** Geometric SIFT

**HWD:** Head Worn Display

**LoG:** Laplacian of Gaussian

**NCC:** Normalised Cross Correlation

**OS:** Operating System

**PCA-SIFT:** Principal Components Analysis – Scale Invariant Features Transform

**RBF:** Radial Basis Function

**ROI:** Region of Importance/Interest

**SIFT:** Scale-Invariant Feature Transform

**SURF:** Speeded-Up Robust Features

**SUSAN:** Smallest Univalued Segment Assimilation Nucleus

**SVD:** Singular Value Decomposition

**2D:** Two Dimension(al)

**3D:** Three dimension(al)

# CHAPTER ONE

## Project Overview

### 1.1 Aims and Objectives

The aim of this work is to correct geometric distortion observed when undistorted images and video frames are projected onto both static and dynamic/deformable non-planar projection surfaces or displays. The work features a projection system consisting of a projector projecting images and video from a computer onto the projection surface, a single uncalibrated camera that monitors the projections on the surface, and the computer providing image processing and system control.

Homography Shaping is a term used in this work to describe the derivation of the shape of the distortion-corrected projection. Through homography shaping, the desired observed projection is made to look like the projection was done on a planar surface. The homographies or 2D transformations between the image being projected, the desired image on the projection surface, and the camera image of the projection are defined in Section 3.3 and the distortion correction process through homography shaping is explained in Section 4.3.

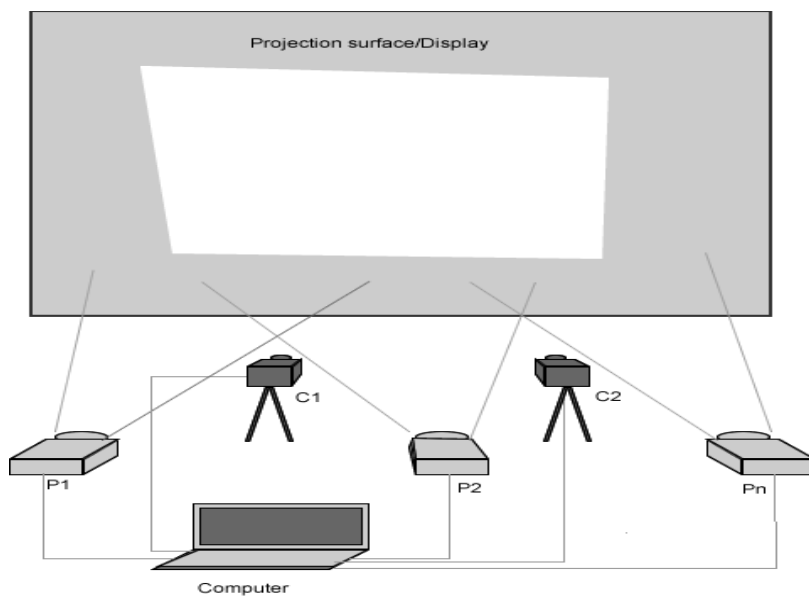
Image Warping is generally a transformation that changes the spatial configuration of an image [1]. In this work image warping is the process of transforming the spatial configuration of the image being projected in order to observe its non-distorted projection on the non-planar projection surface. The term *pre-warping* is also used in this work to refer to the warping of the image being projected for the same purpose. The simple-to-apply Radial Basis Function (RBF) interpolation is used to implement the warping of images.

The objectives set out to be met include the following:

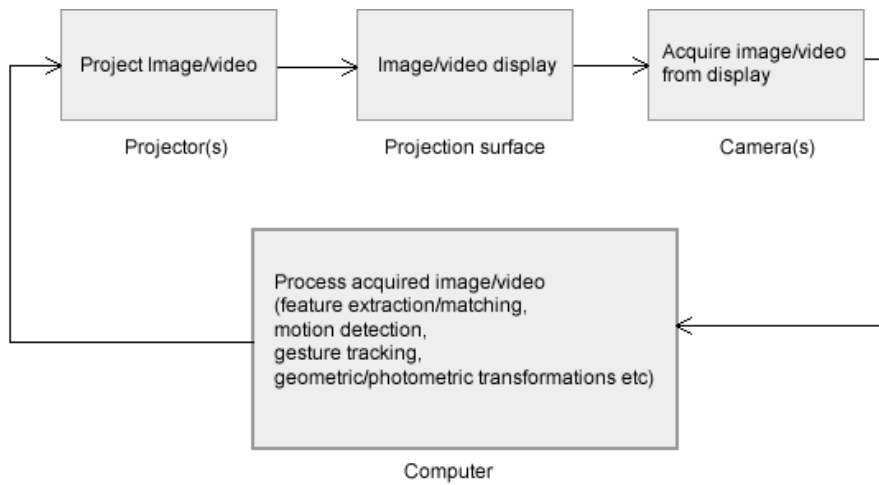
(i) The distortion-correction system developed will work automatically without needing a calibration image or structured light pattern to first estimate or measure the level of distortion on observed projections brought about by the non-planar surface.

(ii) The distortion-correction system will be able to operate in real-time in order to be used to correct distortions in practical projection systems including optical camouflage systems deployed in vehicles and computer-simulated golf systems, both described in the following paragraphs.

The arrangement of a projection system using cameras for feedback and computer for processing and control is shown in figure 1 while the block diagram describing the system functionality is shown in figure 2.



**Figure 1: The components of a projector-camera system consisting of multiple projectors (P1, P2 ... Pn), two cameras (C1, C2), a computer and projection surface/display.**



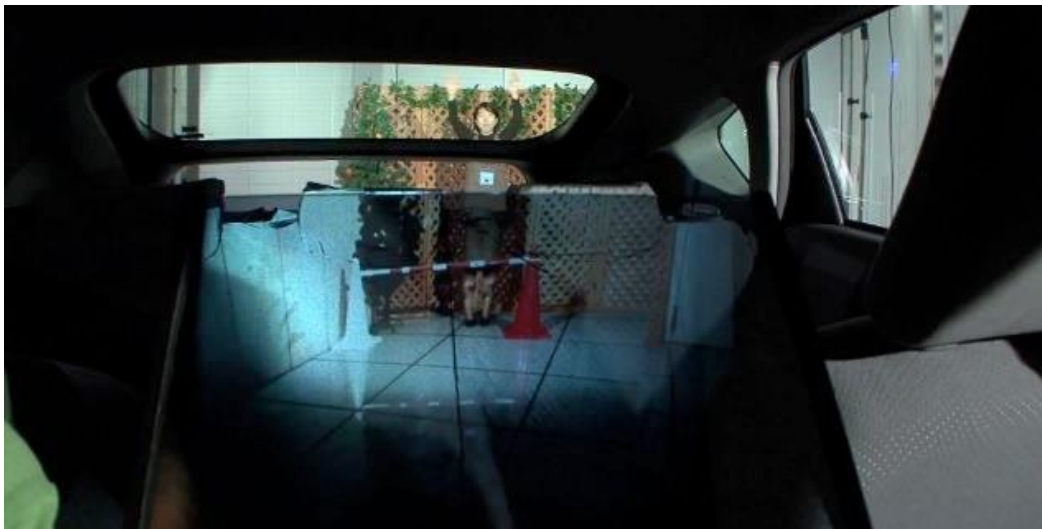
**Figure 2: Block diagram of a projector-camera system showing its components and their functions.**

Conventional standalone projectors are used to provide visual content from computers onto planar or slightly curved surfaces to audiences. These projection systems do not introduce noticeable geometric distortion to the observed visual content as straight projected lines and curves are viewed respectively as straight lines and curves with perspective transformation brought about by the relative position of the projector, projection surface and viewpoint of the observers.

Apart from delivering presentations to audiences, projectors have been used in a variety of ways to either enhance the functionality of some existing systems or form integral parts for others. For example in a prototype system described in [2] a projector is used to deliver structured light pattern to a laparoscope (a surgical device for viewing internal structures of the body) to aid in the acquisition of 3D video data of a surgery site. Through an enhanced Head Mounted Display the user can perform operations having 3D video data from both the enhanced laparoscope and an augmented video source. Projectors are used to project images onto real objects to enhance their appearance and make them more compelling to viewers. In such systems [3] video cameras are used to acquire projected images and images of the environment for system calibration and measurement of the environment in order to provide seamless registration of images on the target objects from multiple projectors. Similarly, the authors in [4] describe a projection system that changes the appearance of objects by making them look like other objects. With the images of the target object

acquired by the camera, a compensation image is derived from photometric properties of the object, its environment, and that of the camera and projector.

An optical camouflage system [5] allows users 'see through' opaque objects like the rear seat of a vehicle. For its vehicular application, the system consists of video cameras mounted at the back of the car, a projector mounted behind the front seats, and a rear-facing half mirror mounted above the projector. The projector projects the portion of the view of the outside of the car provided by the camera but not visible to the driver onto the half mirror which reflects the projection onto the back seat. This results in a complete view of the rear of the car.



**Figure 3: Optical 'see-through' camouflage system applied to the backseat of a vehicle.**

Computer-based golf simulators enable golfers to practice golf indoors. They consist of a loose screen attached to a housing onto which the image of a fairway or part of a golf course is displayed by a projector, arrays of sensors (usually infrared) to sense the movement of the ball from the tee to the screen after being struck by the player, and a computer which collects and processes the ball-tracking signals and performs other image processing and control functions for the system [6]. The projector also projects the computer-generated video of the flight of the ball through the section of golf course. The illustration of a golf simulator is given in figure 4.





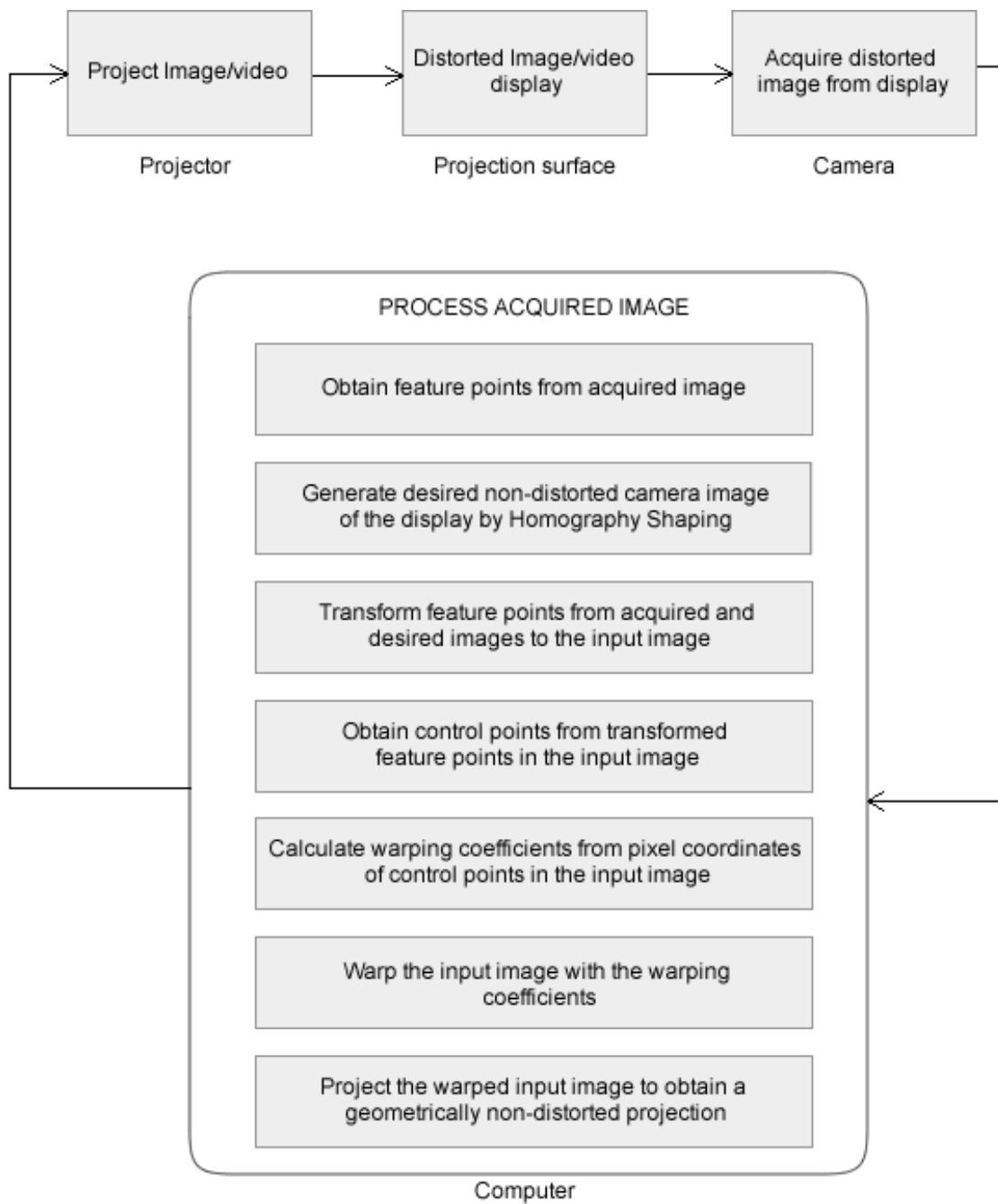
**Figure 4: An *aboutGolf Ltd* golf simulator showing its projector, loose screen, and tee area [7]**

Projecting images and video on non-conventional (for example non-planar, coloured and/or textured) projection surfaces may introduce undesirable forms of distortion. Geometric distortion whereby shapes of projected images become altered to the observer would occur on the non-planar backseat of a car having the optical camouflage system (figure 3) and on the non-planar simulated golf screen as it is struck by the golf ball. Radiometric distortion whereby projected colours change as a result of the mixing of colours between the projection and the projection surface would also occur in both systems. Both forms of distortion need correcting for better user satisfaction and many researchers have developed various approaches to mitigating them. A discussion of the most relevant distortion-correction approaches is presented in the literature review of Chapter 2.

## **1.2 Methodology**

The block diagram describing the connection of components and highpoints of the processing done by the computer in correcting geometric distortion for projected displays is given in figure 5. An elaborate description of the whole process is given in Section 4.3. The process starts by projecting an image or video from the memory of the computer onto the non-planar surface and obtaining a snapshot of the projection with the camera. This snapshot represents the image seen by the observer. Next, feature points including edges and corners of the image obtained by the camera are detected. The detected feature points

are then used to measure the amount of geometric distortion in the image. A significant amount of distortion informs the choice of the decision of the system to carry out the distortion correction and the appropriate image warping direction. Next, the feature points are used to construct the desired non-distorted image (the standard) and obtain a 2D homography between the camera image and the projected image (the input image). Transformation of the same feature points from the camera image to the projected image yield control points. These control points are used to calculate the RBF coefficients needed to warp them in the predetermined image warping direction. The RBF coefficients so obtained are then used to warp all the pixels of the projected image. The system then re-projects this warped image to give a non-distorted view of the projection. An image of the distortion-corrected projection may be obtained by the camera and compared with the standard to measure the effectiveness of the correction process.



**Figure 5: Components of a geometric distortion-correction system and process**

### **1.3 Novelty of the Distortion-correction Approach**

This work has a number of novelties when compared with some other distortion-correction systems discussed in Section 2.3. The individual novelties are presented in Chapters 4 and 5 and include:

(i) This work presents a geometric and algebraic explanation on how geometric distortion occurs when projecting an image from a plane to a non-planar surface in Section 4.2. The derivation of the direction of warping of the projected image is also given. This treatment of the distortion generation and correction process for projected displays can be developed further into a standard distortion-correction simulation tool for future applications.

(ii) The distortion-correction process for a dynamic system with a deformable projection surface whose shape changes with time presented in this work is monocular, does not require the calibration of neither the projector nor the camera, and does not require the 3D measurement or construction of the projection surface. It is therefore a less costly alternative in terms of system components and processing requirements than the calibrated binocular dynamic distortion-correction system described in Section 2.3.

(iii) By taking advantage of the nature of the projection surface, important assumptions on the distortion pattern of the non-planar projection surface are made in Section 4.5.3 of this work. The resulting distortion-correction system developed does not therefore require the projection and detection of any structured light pattern or calibration image on the projection surface as some other systems like [8] do. The need for developing imperceptible patterns and detection algorithms for them is also avoided. Imperceptible patterns are very difficult to extract because of the radiometric modulations that the projection surface and noise add to the camera-captured image. The distortion-correction system developed in this work can therefore be used in static and dynamic projection systems without any interruptions.

(iv) A much simpler RBF interpolation approach to warping as opposed to other processing-demanding warping methods like Bezier patches used in other systems is presented in this work. The optimisation of the RBF image warping algorithm yielding a time saving of 99.32% (Section 5.6) for real-time operation is also presented in this work. Practical applications include possible presentations in places where there is no planar projection surface. For viewers to enjoy a distortion-free view they should have similar field of view as the camera used to observe the surface because the distortion correction is carried out through the

viewpoint of the observing camera. A method by which distortion correction may be achieved for more than one view is given in Chapter 6. In simulated golf practice the user plays the golf ball against a deformable surface that dampens the motion of the golf ball and prevents injury to the player as the ball rebounds off the surface. The trajectory of the golf ball and some text telling its measured speed may also be displayed on the surface. The player's experience will also be improved by removing the geometric distortion from the entire projection as the ball strikes the projection surface.

#### **1.4 Organisation of Thesis**

Chapter 2 is a combined review of already-developed techniques for solving image distortion problems in some areas of computer vision including projector-camera systems. It also includes popular image warping techniques.

Chapter 3 presents the theories behind all the projective transformations and image processing techniques used in this work. It discusses the 2D image homography transformations, popular techniques for detection of feature points in images, various methods of warping an image with illustrations, important concepts of straight-line geometry and its application in measuring geometric distortion in an image, and various methods used in matching or comparing images that give an indication of how much the geometries of a set of images differ from each other.

Chapter 4 presents the techniques developed in this work to correct geometric distortion for static projection displays. It begins by explaining the cause of distortion when a straight line is projected onto a non-planar surface and develops a framework on how this sort of distortion can be solved by warping or changing the shape of the original projected line. It also presents a method to evaluate the distortion/straightness of 'straightest' lines in an image in order to create a basis to compare the distortion values of detected lines in real images.

It combines the warping framework developed earlier with image feature point detection methods, 2D image transformations and the RBF image warping method presented in Section 3.6 to solving geometrical distortion problems in actual projected displays.

Geometric distortion-correction techniques presented in Chapter 4 include one that relies on a calibration image and others that do not require its use. The results of various tests performed with the projector-camera distortion-correction system are also presented.

Chapter 5 extends techniques developed in Chapter 4 to correcting geometric distortion problems in deformable projected displays. It discusses ways by which image processing algorithms used for the correction process can be optimised for real-time operation. It also presents various tests and results to validate these techniques including their application to simulated golf, where the golf system is simulated by striking the projection surface with a projectile.

A general discussion of the whole work including techniques developed and the results obtained is given in Chapter 6. Recommendations for improvement of this work concludes the chapter. Image processing methods not elaborately mentioned in the main body of this thesis are presented in an appendix.

# CHAPTER TWO

## Literature Review

### 2.1 Introduction

In this chapter a survey is carried out to identify the important aspects of this work: (i) the prevailing problem of geometric distortion in images acquired by a typical computer vision system and (ii) the techniques developed to correct it. In Section 2.2 the concept of geometric distortion in images is defined and a survey on its causes and methods developed by researchers to correct it is presented. The focus of this work is correcting geometrically-distorted projections from images of the projection acquired by a camera. Section 2.3 therefore gives a survey of similar projector-camera systems. The survey discusses the projector-camera system calibration and various techniques developed to correct similar geometric distortion problems. Techniques developed to correct geometric distortion on projected displays involve a number of parametric and non-parametric geometrical image transformation or warping techniques. A brief description of these warping techniques is given in Section 2.4. A more detailed description of specific image processing techniques with their underlying mathematical expressions is given in Chapter 3. Image processing techniques used to measure the similarity between images are presented in Section 2.5. A summary of the main discussions in this chapter is presented in Section 2.6.

### 2.2 Image Geometric Distortion: Causes and Correction

Geometric distortion in an image occurs when the geometry of objects in the image differ from the geometry of respective objects in the scene, in ways that the intended purposes of the vision system, such as measurements or object recognition tasks may not be able to be performed because of the distortion. Geometric distortion may present itself as straight lines from the source scene appearing as curves in the image caused by the lens of the camera, the change in perspective such that relative dimensions of objects in the image are not represented as they are in real life because their respective shapes become skewed, objects in the image appearing blurred due to the motion of the camera or the objects in

the scene, or the occlusion of important objects in the image by unwanted objects in the scene may occur.

Geometric distortion occurs in various fields that utilise imaging systems to acquire and analyse images and it may be caused by any of three elements of imaging systems: (i) the nature of the scene or image source, (ii) the medium through which light rays travel to get to the image acquisition device, and (iii) properties of the target acquisition device (a camera for example). The following paragraphs describe various forms of geometric image distortions caused by the various elements of imaging systems and techniques developed to correct them.

The source of the imaging system is the scene object being photographed or studied. Various image sources are encountered in computer vision. In [9] geometric distortion is caused by the non-planar or ruffled paper documents from which images are acquired. To solve the distortion problem, the 3D shape model of a document is constructed from a smooth shading image acquired by Digital In-painting and a 2-pass Shape-From-Shading technique, then flattened to restore the shape of the document to its assumed pleasant planar view. As stated in [9], this type of geometric correction can be used as a preliminary restoration step in Document Image Analysis (DIA) systems. Several other methods to correct the geometric distortion of images of documents caused by the non-planar shape of the source objects have also been developed in [10], [11], and [12]. In [10] for instance, the surfaces of the documents whose images were rectified were modelled as cylindrical surfaces. The geometrical rectification of the images was achieved by estimating the extent of bending of the document surfaces using the mapping of points from the scene to points in the image and using this information to digitally lay the image out to make it appear flat.

A typical area where the transmission medium causes geometric distortion of acquired images is underwater imaging. Distortion of the image of an object in water may be caused by the rapid flow of water in the scene, image noise caused by floating particles, or motion blur or refraction of light waves at the water-air interface.



In [13] it is assumed that geometric image distortion is only caused by dynamic refraction at the water surface and presents itself as motion blur in the acquired images. The dynamic refraction was caused by both unidirectional cyclic waves and circular ripples. The distortion-correction or 'deskewing' process therefore involved correcting distortions caused by both kinds of water waves. While carrying out 3D measurements of objects in water using a fish-eye stereo camera, [14] used a ray-tracing method involving modelling the refraction of light from air to the water surface by Snell's law to account for the geometric distortions present in the stereo images. On the other hand, by utilising the distortion phenomenon caused by the refraction of light rays through an air-water medium differently, a method of reconstructing the surfaces of transparent non-rigid bodies like water by measuring the geometric distortion of objects lying underneath them was developed in [15]. This method first models the apparent movement of the object underneath the transparent surface using optical flow, extracts the optical flow parameters and then reconstructs the surface using the estimated surface normals.

Geometric distortions in images introduced by the target camera can either be caused by the camera lens or sensor. Lens distortion is caused by the spherical shape of the camera lens and it manifests as straight lines in the scene appearing as curved lines in the image. Lens distortion can be grouped into 3 types according to [16]: (i) shift of optical centre which occurs when the camera sensor is shifted in the optical plane, (ii) radial distortion which includes barrel distortion when straight lines appear curved and bend away from the centre of the image or pincushion distortion when straight lines appear curved and bend towards the centre of the image, and (iii) decentering distortion, which occurs when the fixation of the lens is not orthogonal with respect to the image sensor and optical axis thereby causing image distortion in a direction that is tangential to radial distortion. The more prevalent radial lens distortion has been modelled in various ways in reducing its effects in imaging systems and it is essentially part of the camera calibration process. Examples are seen in [17], where distortion in the two image directions is approximated to a 2<sup>nd</sup> degree polynomial in  $r$  the radius of distortion, and [18] where higher-degree polynomials are obtained and solved through a covariance line-rectification method.

## **2.3 Projector-Camera Systems**

Cameras are incorporated into the computer-projector loop (figures 1 and 2) to monitor the display of a projector. Such systems may also be called smart projection systems because of the automatic problem-solving functions they perform. A projector-camera system consists of one or more projectors, one or more cameras, and one display surface either planar or curved, plain or patterned, or consisting of a combination of a number of surfaces. The purpose of the camera is to monitor what is being projected onto the display surface by the projector by capturing projected images. Further processing can be done on the captured images by appropriate software running on a computer. This processing may involve projector-camera system calibration, radiometric and geometric distortion correction of the projected images, shadow elimination from the display surface [19], or creating a seamless arrangement on the display surface of multiple images from multiple projectors.

### **2.3.1 Projector-Camera System Calibration**

Projector-camera system calibration involves estimating the intrinsic and extrinsic parameters of the projector and camera or deriving their respective calibration matrices. Projectors and cameras are usually modelled with the same calibration parameters as the pinhole camera. Various methods of calibrating a projector-camera system exist depending on what is to be achieved in each application. Examples are seen in [20] and [21] where the projector-camera system is calibrated for 3D measurements and modelling respectively. The two-step calibration method described in [20] estimates the intrinsic and extrinsic parameters of the camera and projector. A structured light pattern or calibration image consisting of a grid of black squares on a white background is pasted on the bottom half of a planar projection surface. The 3D coordinates of the corners of the square with respect to the coordinate frame of the projection surface are known. The projector is used to project a calibration image consisting of the same gridded pattern on the top half of the projection surface. Several images of the patterns on the top and bottom halves of the projection surface are captured by the camera at different poses. Correspondences between the coordinates of the feature points of the pasted pattern on the bottom half of

the projection surface and the feature points on the camera images are used to estimate 2D homographies between the camera planes and the projection surface. These homographies are then used to estimate the intrinsic and pose parameters of the camera. The 3D coordinates of the feature points of the projected pattern on the top half of the projection surface are estimated by back-projecting the 2D feature point coordinates obtained from the camera images using the camera calibration parameters so obtained. The correspondences between the coordinates of the feature points on the projected image and the back projected ones are used to estimate 2D homographies between the projector planes and the projection surface. Finally these homographies are used to estimate the intrinsic and extrinsic parameters of the projector.

The calibration method described in [21] first uses the coordinates of corner points in camera images of a checkerboard pattern pasted on the projection surface to estimate the calibration parameters of the camera. It then uses the coordinates of corner points in images of a ray code pattern projected onto the checkerboard pattern to obtain the parameters of the projector. This calibration method also estimates radial lens distortion parameters for both the camera and projector.

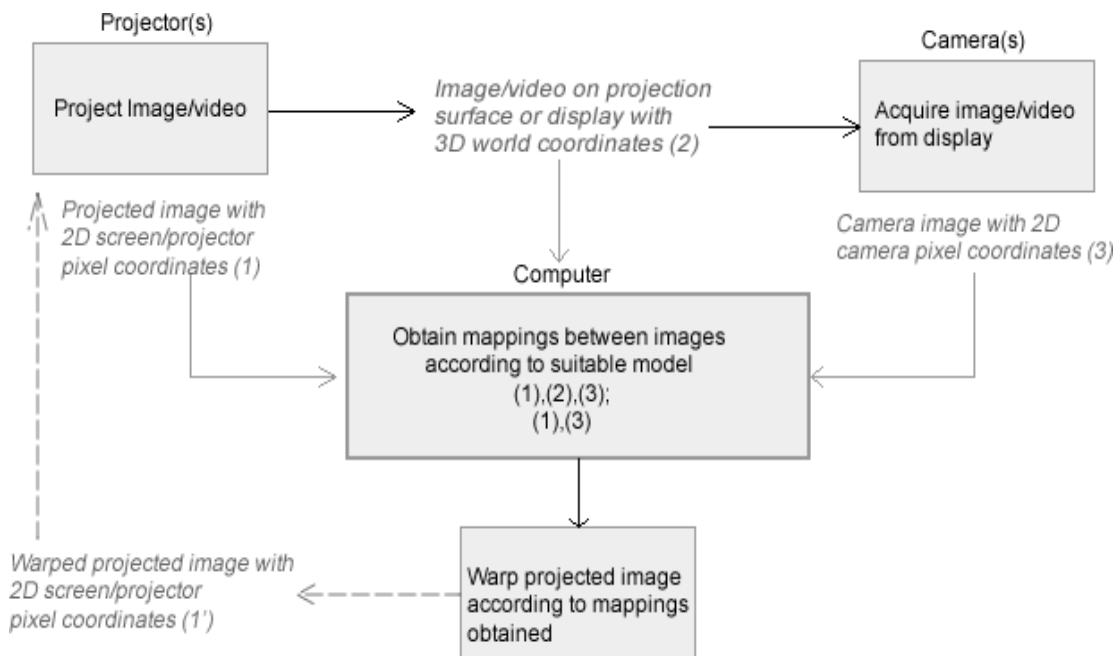
A radiometric calibration of a projector can also be carried out to compensate for colour differences between the image being projected and what is observed on the display surface. An example of this kind of calibration is seen in [22] where the fast colour compensation is achieved by colour filters attached to the projector.

### **2.3.2 Correction of Geometric Distortion for Projected Displays**

When projectors are used on flat or planar surfaces and the plane of projection is not at right angles with the principal axis of the display surface, rectangular shapes lose their squareness and appear distorted. Straight lines however remain straight. This type of distortion known as keystone distortion [23] is caused by the pose of the projector with respect to that of the projection surface. Several methods have been developed to automatically correct keystone distortion and two of them are reported in [23] and [24].

With such applications as virtual and augmented reality or the Office of the Future and Transportation of the Future, it has become imperative to have projectors display content on surfaces that are non-planar or undulating. This kind of projection display however comes with a problem: the surface topography causes a shift in the direction of incident and reflected light, causing the observer to observe geometrically distorted images. Several techniques of correcting geometric distortion on projected displays have been developed over the years. All of these correction techniques involve the drawing up of mappings or correspondences between at least two of three images involved in the projection: (i) the projected image whose coordinate system corresponds to pixel coordinates of the generating computer and projector (ii) the image appearing on the display or projection surface with coordinates corresponding to a world coordinate system, and (iii) the image of the projection with coordinates corresponding to the pixel coordinates of the observing camera. The required mappings are obtained by applying a suitable parametric model to corresponding features of the respective images and obtaining the parameters of the model.

A generalised block diagram of the distortion correction process common to these techniques is shown in figure 6. Some techniques rely on easily-detectable features of a calibration image or structured light pattern projected onto the non-planar projection surface in order to apply the parametric model but others do not. The mapping parameters obtained are then used to transform the projected image in order to obtain a non-distorted view of the projection. The following discussion divides techniques developed to solve the distortion-correction problem into two groups depending on whether or not they use a calibration image or structured light pattern in the distortion-correction process.



**Figure 6: Generalised block diagram describing the distortion-correction process for projected displays**

In [25] the structured light pattern used to generate piecewise linear forward and reverse mappings between points on the projected image and the camera image of the projection consists of horizontal and vertical sets of alternating black and white stripes. The system makes use of a single uncalibrated projector and obtaining the mapping between the images does not require a full 3D reconstruction of the projection surface. Although the authors presented results for structured light patterned images, they did not mention how the system would deal with dynamic projection systems.

In [26] a projection system consisting of 4 to 8 projectors and a camera is used to display an image on non-planar surfaces to create a consistent appearance across a large zone of viewing locations in 3D space. The non-planar surfaces include cylindrical, conic and intersecting planar surfaces. A 2D mesh-based method is combined with 2D lookup tables to derive the geometric mappings from the image being projected to the projection surface and from the image of the projection to the projection surface. In this method, checkerboard strips are attached to the borders of the projection surface to surround the

projection. These strips are used in the detection of important regions of the projection surface in the camera image and the subsequent derivation of geometrical mappings between the images.

The authors in [27] designed ways of inserting imperceptible patterns into the projected image to aid the detection of image features needed to develop geometric mappings between the projected image and the camera image of the projection. The system makes use of a calibrated camera and projector and requires the synchronisation of both devices in order for the system to function properly. A synchronisation pulse provided by an external trigger initiates the projection and capture of the embedded pattern by the projector and camera respectively.

A distortion-correction system that relies on the projection of video frames whose luminance have been altered according to a coded image composed of many pattern (or shape) primitives is described in [28]. These luminance-altered video projections are captured by a carefully-synchronised camera. The coded pattern primitives are recovered by decoding the result of subtraction of two consecutive frames. Next, the decoded patterns are matched with a reference code to determine the visible area of the projected frame and the geometric mapping between the projected and captured images and the projection surface. The following input frames are then warped appropriately to correct the distortion on the projection surface.

The authors of [29] developed a simple auto-geometric correction method for non-planar projections from one or more projectors using an uncalibrated camera and a cylindrical surface. The process involves projecting a checkerboard pattern on the screen, capturing the projection with the uncalibrated camera and then detecting features (checkerboard pattern corners) of the camera image. The detected features on the camera image are then mapped to the corresponding features on the projected image using a rational Bazier patch. A region on the surface where the corrected projection should appear on the camera image is defined by a rectangle and a similarity transformation between the input image and this target rectangle is obtained. The similarity transformation is used to transform detected

image features to corresponding positions on the projected image. The Bezier patch transformation  $R$  is then used to transform these corners to obtain a warped image which when projected onto the screen produces a geometrically-corrected display. They proposed an extension of their method to accommodate more than one projector whereby the geometric registration of each projector is carried out one at a time. They suggested using intensity blending techniques to remove unwanted intensity fluctuations between each projection.

In [30] a markerless view-independent registration of multiple distorted projections on extruded surfaces using an uncalibrated camera by creating a pasted wallpaper display with the respective image projections is described. A cylindrical vertically extruded surface was used. Two important priors were imposed on the surface: that the surface is formed by a 2D curve and that the aspect ratio of a planar rectangle formed by four corners of the surface is known. The method is reported to ensure a geometric registration of the projected imagery devoid of stretching or distortion by parameterising the display surface in 3D rather than in the 2D camera image space as described in [26]. Assumptions were also made on the world coordinates of the corners and edges of the rectangle and an implied vertically-symmetrical display surface. They then used a single camera image of the display, the focal length of the camera extracted from the Exchangeable Image File Format (EXIF) tag of the image format and its sensor size, and the known aspect ratio of the surface to estimate the intrinsic and extrinsic parameters of the camera by a non-linear optimisation method. The estimated parameters of the camera were then used to back-project the top and bottom image curves to obtain estimates of the 3D top and bottom profile curves of the display surface. The next task involved developing a mapping from the projector to the display surface via the camera. By projecting a pattern from each projector onto the display surface, rational Bezier patches were used to define mappings from the projected images to the camera images of the projections. Each patch was back-projected from the camera image to obtain its 3D surface point and subsequently its 2D Bezier patch parameter. The area of the display spanned by each projection was therefore defined in a single camera image.

Various distortion-correction techniques that do not rely on the projection of calibration images or structured light patterns to obtain geometric mappings between the projected image, the image on the projection surface and the camera image of the projection have been developed. Most of these methods however require that the camera(s) and projectors used in the systems are calibrated either online during the projection or offline in a separate projector-camera calibration stage. The authors in [31] for example developed a method of correcting the geometric distortion of images projected on a non-planar surface using a previously-calibrated projector and a previously-calibrated camera. The calibration process of both devices however uses a calibration image as described in [32]. This calibration stage is used to estimate the fundamental matrix of the projector and camera. The estimated fundamental matrix enables point correspondences between the projected image and the camera image of the projection to be found by tracing corresponding epipolar lines from both images to their 3D points of intersection on the projection surface that is modelled as a B-spline surface. With point correspondences between the projected image with the projection surface, and the camera image of the projection with the projection surface respectively known, the point correspondences between the camera image and the projected image can be found. These correspondences are used to update the projected image to correct geometric distortions observed by the camera and the viewer. The success of this method depends on: (i) fixed positions of the camera and projector throughout the process, (ii) the accuracy of the B-spline surfaces in approximating the shape of the projection surface, (iii) the use of a calibrated camera-projector pair, and (iv) the use of structured patterns for calibration of the camera and projector.

The authors of [8] describe a geometric correction system for projected displays using calibrated stereo cameras. The system is reported to adapt itself to changing shape or position of the display surface and eliminates the use of structured light patterns. This system does the correction in 3 steps. First, the perspective projection matrix of the projector is estimated from point correspondences between the projected (input) image, and the pair of images obtained by the stereo camera. This projector calibration process involves the use of Scale-Invariant Feature Transform (SIFT)-based feature matching to obtain correspondences between the projected image and one of the stereo camera images.



Random Sampling and Consensus (RANSAC) is used to eliminate such distorted image points that pose as outliers. With Phase-only Correlation (POC), correspondences between the same points of the stereo camera images correctly matched by SIFT are found, leading to a complete feature-point correspondence between all three images. Second, a dense high-accuracy 3D measurement of the display surface is carried out using phase-based correspondence matching to correct the geometric distortion of the projection. During this step, the rectified images of the stereo camera are obtained from the camera calibration parameters and correspondences between points on both images are found. Third, the 3D coordinates of the points on the projection surface are back-projected to the projected image to create a mesh. The mesh is used to transform the projected image to obtain a geometrically-correct projection on the projection surface. Avoiding 3D surface reconstruction of the display surface is possible because the warped image at the input of the projector and the image obtained by the camera are 2D and can be modelled by a 2D projective transformation. This idea is also supported by [33] with static screen and viewpoint but argue that 3D reconstruction of the screen and dynamic warping are inevitable when the viewpoint changes.

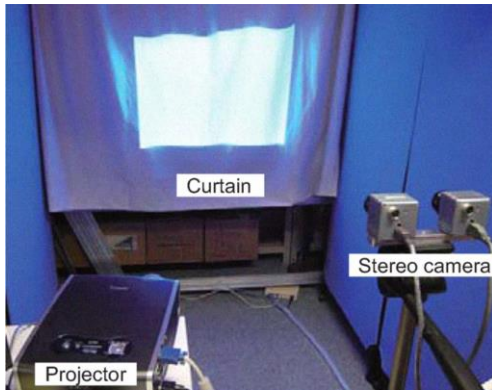
### **2.3.3 Correction of Geometric Distortion for Projected Displays Developed in this Work**

In this section comparisons are made between the distortion-correction techniques discussed in Section 2.3.2 with the technique developed in this work. First, the distortion-correction technique developed in this work functions automatically, without the need for first projecting a calibration image or structured light patterns as reported in [25], [28], and [29]. In situations where the shape of the projection surface changes or the pose of the projector or/and the observing camera change, the projection in these systems has to be interrupted to allow for the calibration image to be reprojected onto the surface in order to maintain a distortion-corrected view of the projection. The technique developed in this work can therefore work without interruptions for projecting video frames and in situations where the projection surface shape and poses of the observing camera and projector do not remain constant.

Second, the technique developed in this work uses the homography between natural features of the projected image and the camera image of the projection to work out the transformation of the projected image to obtain a non-distorted view of the projection. This technique can therefore work with a single uncalibrated camera and projector as opposed to distortion-correction techniques described in [8], [27], and [31]. The use of the 2D homography also eliminates the need for estimating and reconstructing the 3D projection surface as seen in [31], and [30]. Eliminating extraneous operations in the distortion-correction process makes the method developed in this work capable of performing distortion-correction in real-time.

Third, none of the distortion-correction techniques discussed in this section presented results for deforming surfaces or changing projector and camera positions. A few authors however mentioned the capabilities of their techniques to handle such situations without any results to support their claim. The technique developed in this work to cope with such situations is given comprehensive treatment in Chapter 5.

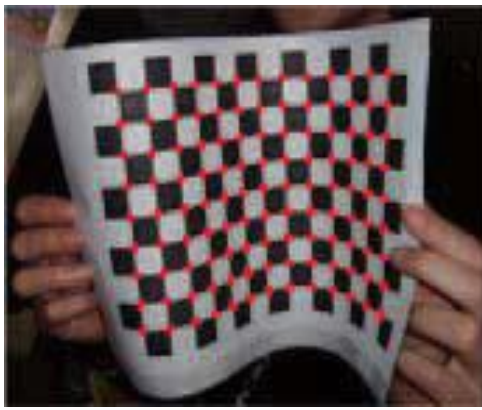
It is important to note that non-planar projection displays or surfaces used by the various authors for testing include smooth cylindrical surfaces, hung curtains and intersecting planar surfaces. Tests performed in this work were done using a hung curtain and intersecting planar surfaces as well. Results presented in Chapters 4 and 5 show significant improvement from distorted to distortion-corrected projections for the same surfaces. Some of the projection surfaces used in the literature discussed in this section are shown in Figure 7.



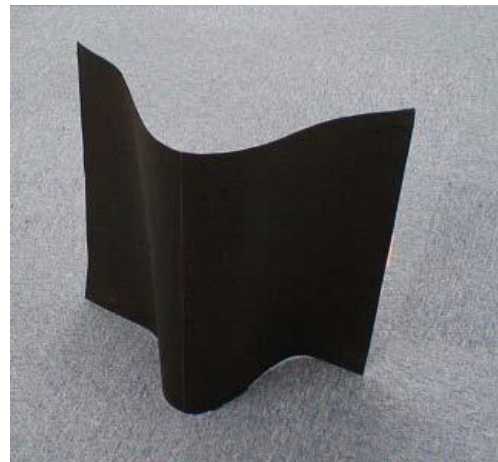
(a) Hung curtain [8]



(b) Cylindrical projection surface [26]



(c) Warped piece of paper [29]



(d) Warped projection surface [31]

**Figure 7: Various projection surfaces used to test distortion-correction techniques mentioned in literature**

## 2.4 Image Warping Techniques

Image warping or deformation is described by [34] as the transformation of the image plane to itself, with the grey values of the image transformed according to the transformation of their respective coordinates. Image warping has been used for geometric image distortion correction and image registration tasks whereby one image is warped to align it with another image. When an undistorted image is projected onto a non-planar surface, the irregularity of the surface would result in a geometrically-distorted view of the image. The observed pattern of distortion of the image depends on the profile of the projection surface. To restore a non-distorted view of the projection, a warping function can be used to pre-

distort the input image before projection. Image warping techniques can be grouped into parametric and non-parametric transformations [35]. Examples of parametric transformations include: (i) Translation where the only difference between the original image and the warped image is a displacement along one (horizontal or vertical) or both directions, (ii) Procrustes Transformation which is a four-parameter transformation causing scaling, rotation, and translation in the horizontal and vertical directions, (iii) Affine Transformation which is a six-parameter generalisation of the Procrustes Transformation [35] allowing stretching, shearing and translation along both image directions, (iv) Perspective Transformation, an eight-parameter transformation typical of the appearance of an image being viewed from a fixed point in space, (v) Bilinear Transformation which also has eight parameters, generalises the affine transformation but invariant to rotation, and (vi) polynomial transformations including quadratic, biquadratic, cubic and bicubic as examples. Parametric transformations for image warping have been observed to perform poorly with local distortions [36].

Non-parametric image warping techniques are more localised techniques that are derived from matching interest points (also referred to as control points or landmarks) in an original image and corresponding points in a warped version of the image. Having known the correspondence between these points, a function is sought which transforms the interest points exactly from their initial to final positions and estimates the transformation of all other points in the image [37]. This is an interpolation process and the function is known as an interpolating function or interpolant. Smoothness constraints are imposed on the interpolating function [35], with each parameter of the function having only a local impact on the shape of the warped image [38]. Examples of non-parametric image warping techniques include 2D and 3D splines specifying curves and their respective patches specifying 2D and 3D surfaces, Bezier curves, double quadratic curves, B-splines, rational B-splines, non-uniform rational B-splines (NURBS) and thin-plate splines [39]. Radial basis functions (RBFs) and a Bayesian approach offering further possibilities for specifying smoothness constraints are also examples of non-parametric image warping techniques. Rational Bezier patches have been used in [40] to model radial and tangential distortions of the projector lenses while calibrating planar multi-projector displays. The registration of

images of the human pelvis with large distortion using thin-plate splines is mentioned in [41] and used for 3D face registration in [42].

An RBF is a sum of basis functions whose respective values depend on the distance of the point of evaluation and the interest points plus an affine transform. RBFs have been effective for multivariate interpolation of scattered data [34]. The use of RBFs to model free-form surfaces in optics design and applied to Head Worn Displays (HWD) is described in [43]. In [44], RBF interpolation is used to correct for distortion created by the optics of an off-axis eyeglass display, (a type of HWD). The distortion correction process involves using RBFs to pre-warp input images to be displayed on the device's microdisplay in a way that counteracts the distortion introduced by its optics to give the user an undistorted view of the input image. While working to design cranial implants for the repair of defects in the skull, [45] used RBFs to interpolate incomplete surfaces derived from depth maps of CT scans of the skull. The choice of use of RBFs for this purpose as against other methods like global and piecewise polynomial fitting, include: (i) RBFs adaptability to interpolate scattered data even with large data-free regions, (ii) the relative ease of obtaining an inverse mapping with RBFs, and (iii) the fact that RBF interpolation does not require that known data points be in a regular grid. [46] applied compact RBFs with local support to the restoration of images and videos having corrupted pixels. The method relies on the prior knowledge of the actual pixels corrupted by noise. RBF interpolation has also been used for aerodynamic applications as in the gradient-based shape optimisation of an aircraft wing [47]. Since the optimisation process involves deforming computational fluid dynamics (CFD) meshes to map CFD parameters to optimisation cost functions, the authors used RBF interpolation to perform mesh deformation because it can be used on arbitrary meshes and it is capable of saving computational cost. RBF interpolation has been applied to animated deformation of the human face to produce facial expressions by manipulating a number of facial feature or mesh points [48]. The advantages gained by the RBF method include it providing a smooth geometric deformation, eliminating the need for modelling mechanisms like muscles as in the case of Mass-Spring deformation systems, can easily be automated and can work on any mesh without modification.

In this work warping of the projected image through RBF interpolation is done to obtain a distortion-corrected projection as described in Sections 1.3, 3.6 and 4.3.4. The reasons for using RBF interpolation include the following:

(i) It is relatively easy to obtain the coefficients without using any recursive algorithms. All that is required to deploy RBF interpolation in this work is a well-distributed set of control points in the projected image with initial coordinates specifying the positions of these points in the undistorted state and final coordinates specifying their positions in the warped state. The computation time required to warp the projected image is further reduced by as much as 98% by approximating the evaluation of the RBF by linear interpolation as shown in Section 5.5.3.

(ii) Warping the projected image by RBF interpolation has not previously been used to correct distortion in non-planar projected displays and this work presents a good addition to its numerous applications.

## **2.5 Image Similarity Measurement**

Image similarity measurement techniques evaluate how closely two images match with each other. Some image similarity measurement techniques include:

(i) Image normalised cross-correlation (NCC), an area-based method that measures the similarity of two images as a lag of one relative to the other [49]. The better the match of both images, the higher the peaks of the normalised cross-correlation function. Normalised cross-correlation has been used for general template matching, to obtain the location of a template in a larger image [50] [51], matching of MRI images [51], and remote sensing images [52].

(ii) Fourier transform-based image similarity measurement techniques which compare images in the frequency domain. These techniques are preferred to the normalised cross-correlation techniques when acceleration of computation is desired and the images are corrupted by frequency-dependent noise [49]. The phase correlation technique is an example of a Fourier transform-based image similarity measure. It obtains the cross-power

spectrum of the image set by multiplying the Fourier transform of the first image with the complex conjugate of the second image and calculating the inverse Fourier transform of the normalised result. A distinct peak of the phase correlation indicates a good match of the images.

The Normalised Cross Correlation (NCC) is used in tests in Chapters 4 and 5 to evaluate the effectiveness of the distortion-correction process. Its mathematical expression is given in equation 47 of Section 3.7.1. A desired non-distorted camera image of the projection is obtained through homography shaping. This desired image is used as a matching template for the camera images of the distorted and distortion-corrected projections. The increase in the peak NCC resulting from matching the distorted and distortion-corrected images with the template serves as an indication of the improvement of the distortion-correction technique carried out in this work.

## **2.6 Chapter Summary**

The aim of this chapter was to highlight the important aspects of this work including the problem of geometric distortion in images acquired by a typical computer vision system and image processing techniques developed to correct it. The following discussions were presented:

(i) The definition of image geometric distortion, its causes, and the techniques developed to correct it in imaging systems including document restoration, underwater imaging, and solving camera lens distortion.

(ii) A survey of projector-camera systems including system calibration and distortion-correction techniques. The various improvements of the distortion-correction process carried out in this work were also presented.

(iii) A discussion of various parametric and non-parametric image transformation or warping techniques that may be used to correct geometrical image distortion. A description of the RBF warping technique with application areas including modelling of surfaces

encountered in medicine, entertainment, computational fluid dynamics and animation was also presented.

(iv) The presentation of Normalised Correlation Coefficient (NCC) and Fourier transform-based methods as techniques used to measure the similarity between images. A more analytical treatment of the various aspects of this work mentioned in this chapter is given in Chapter 3.



# CHAPTER THREE

## Theory of Image Processing Techniques

### 3.1 Introduction

This chapter presents useful theoretical principles upon which the processing carried out in this work are based. A description of image feature detection techniques including edge, corner and blob detectors is given in Section 3.2. The various image feature detectors were used in this work to enable image registration and matching. The level of geometric distortion in an image can be determined by measuring the deviation of originally-linear image features such as the corners of a calibration image from straight lines. An algebraic treatment of the geometry of the straight line is therefore presented in Section 3.4. Important geometrical principles of 2D projections and the general concept of the 2D homography and its extension to projector-camera systems are presented in Section 3.3. The mathematical description and illustrations of parametric image warping techniques that may be used to solve image distortion problems discussed in Section 2.4 are given in Section 3.5. Section 3.6 describes the RBF image warping technique and derives mathematical expressions for obtaining RBF coefficients to be used in warping projected images in order to correct the observed distortions on projected displays. Section 3.7 discusses image similarity measurement techniques that can be used to evaluate the effectiveness of the distortion-correction process. A summary of the main discussions in this chapter is presented in Section 3.8.

### 3.2 Image Feature Detection Techniques

A feature is an interesting part of an image such as a corner, edge, blob, or line. Changes or discontinuities of luminance values in an image are very important image characteristics as they usually provide information about the extent of features of individual objects within the image [53]. Numerous image feature detectors have been developed and a few of them were used in this work to extract the control points needed to correct geometric distortion of projected displays through image warping by RBF interpolation. The following

subsections describe the theories underlying the techniques used to extract the needed image features.

### 3.2.1 Image Edge Detection

Edges are curves that follow the rapid change of intensity in an image and they are usually associated with boundaries of objects in a scene [54]. Techniques used to perform image edge detection are divided into two categories: (i) gradient-based methods that locate edges by finding maxima and minima in the first derivative of the image function in one or more directions defined by a set threshold (figure 8(b)), and (ii) Laplacian-based methods which locate edges by searching for zero-crossings in the second derivative of the image function (figure 8(c)) [55].

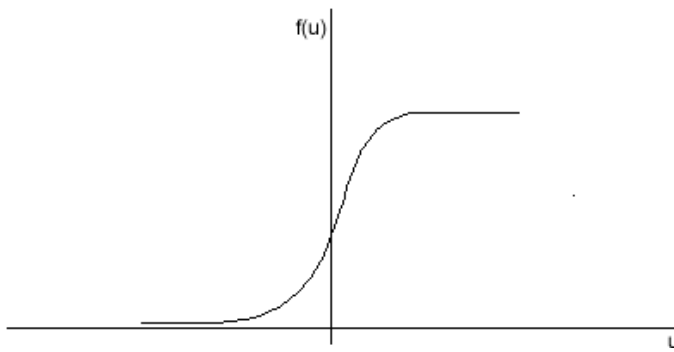
The magnitude of gradient of a 2D image in the horizontal ( $u$ ) and vertical ( $v$ ) directions is given as

$$|G| = \sqrt{G_u^2 + G_v^2} \quad (1)$$

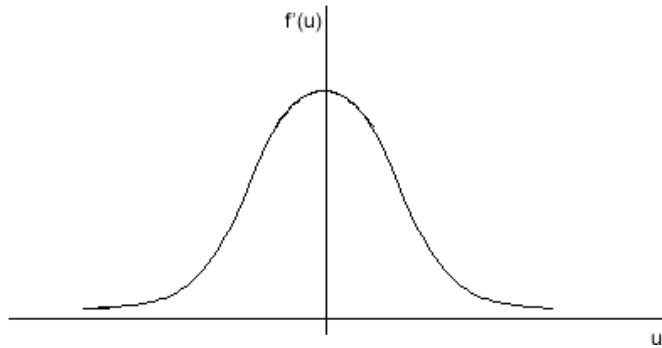
and the orientation of the edge is given as

$$\theta = \tan^{-1} \left( \frac{G_v}{G_u} \right) \quad (2)$$

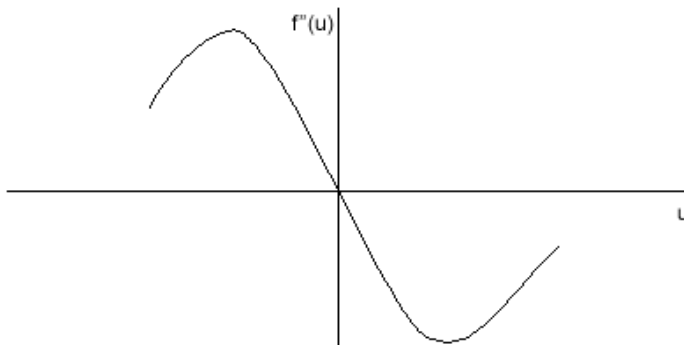
The calculation of the gradients for an image is approximated by finite differences in the respective directions. This is achieved by a set of convolution filter kernels. Examples of convolution kernels are given in Table 1. A gradient image is produced when the kernels are convolved with an input image and a threshold value is applied to the gradient image to obtain the required edges. The output of gradient-based edge detectors is therefore heavily dependent on the value of the threshold [56].



(a) plot of  $f(u)$  showing a rapid change in  $f(u)$



(b) Plot of  $\frac{df(u)}{du}$  showing a maximum indicating an edge



(c) Plot of  $\frac{d^2f(u)}{du^2}$  showing a zero-crossing indicating an edge

Figure 8: Rapid change of univariate function  $f(u)$  indicates an edge and shown by the maximum value of its first derivative  $f'(u)$  and zero-crossing of its second derivative  $f''(u)$ .

**Table 1: Convolution kernels for gradient-based edge detectors**

No.	Name	Convolution kernels																		
1	Sobel	<table border="1" style="display: inline-table; margin-right: 20px;"> <tr><td>-1</td><td>0</td><td>+1</td></tr> <tr><td>-2</td><td>0</td><td>+2</td></tr> <tr><td>-1</td><td>0</td><td>+1</td></tr> </table> <table border="1" style="display: inline-table;"> <tr><td>+1</td><td>+2</td><td>+1</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>-1</td><td>-2</td><td>-1</td></tr> </table>	-1	0	+1	-2	0	+2	-1	0	+1	+1	+2	+1	0	0	0	-1	-2	-1
-1	0	+1																		
-2	0	+2																		
-1	0	+1																		
+1	+2	+1																		
0	0	0																		
-1	-2	-1																		
2	Prewitt	<table border="1" style="display: inline-table; margin-right: 20px;"> <tr><td>-1</td><td>0</td><td>+1</td></tr> <tr><td>-1</td><td>0</td><td>+1</td></tr> <tr><td>-1</td><td>0</td><td>+1</td></tr> </table> <table border="1" style="display: inline-table;"> <tr><td>+1</td><td>+1</td><td>+1</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>-1</td><td>-1</td><td>-1</td></tr> </table>	-1	0	+1	-1	0	+1	-1	0	+1	+1	+1	+1	0	0	0	-1	-1	-1
-1	0	+1																		
-1	0	+1																		
-1	0	+1																		
+1	+1	+1																		
0	0	0																		
-1	-1	-1																		
3	Roberts	<table border="1" style="display: inline-table; margin-right: 20px;"> <tr><td>+1</td><td>0</td></tr> <tr><td>0</td><td>-1</td></tr> </table> <table border="1" style="display: inline-table;"> <tr><td>0</td><td>+1</td></tr> <tr><td>-1</td><td>0</td></tr> </table>	+1	0	0	-1	0	+1	-1	0										
+1	0																			
0	-1																			
0	+1																			
-1	0																			

Each kernel in Table 1 responds to edges in a particular direction. While the Sobel and Prewitt kernels respond to edges in the respective horizontal and vertical directions of the image, the Roberts kernels detect edges occurring at 45° to the respective image directions. The two kernels from a particular set may be applied separately to the image and the output of both kernels combined to obtain the magnitude of the image intensity gradient [55].

The Laplacian is a 2D measure of the second derivative of an image. It is used for edge detection because it highlights image regions of rapid intensity change [57]. The Laplacian of an image with intensity values  $I(u, v)$  is given by equation 3 and usually approximated by the two discrete convolution kernels in figure 9.

$$L(u, v) = \frac{\partial^2 I}{\partial u^2} + \frac{\partial^2 I}{\partial v^2} \quad (3)$$

-0	-1	0
-1	4	-1
0	-1	0

-1	-1	-1
-1	8	-1
-1	-1	-1

**Figure 9: Laplacian Convolution Kernels [57].**

To counter the sensitivity of the Laplacian kernels to noise, the input image is usually smoothed first by a Gaussian smoothing filter before the application of the Laplacian kernel. Since the convolution operation is associative, the Gaussian kernel can first be convolved with the Laplacian kernel to produce a hybrid kernel. The convolution of the hybrid kernel with the input image produces the same result as applying the kernels in the usual order separately. The hybrid kernel or filter is called the Laplacian of Gaussian (LoG) filter. A 2D Gaussian with standard deviation  $\sigma$  centred about the origin  $(0,0)$  of a  $(u, v)$  image coordinate system is given by the expression

$$G_{\sigma}(u, v) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\left(\frac{u^2+v^2}{2\sigma^2}\right)} \quad (4)$$

and a 2D LoG filter obtained from its convolution with a Laplacian kernel is given as [55]

$$LoG = \frac{-1}{\pi\sigma^4} \left(1 - \frac{u^2 + v^2}{2\sigma^2}\right) e^{-\left(\frac{u^2+v^2}{2\sigma^2}\right)} \quad (5)$$

### **Canny Edge Detector**

The Canny Edge Detector is an optimum edge detector [57] that was designed to achieve three purposes [58]: (i) providing good detection with low probabilities of detecting false edges and failing to detect real edges, (ii) providing good localisation by detecting edge points that are as close as possible to the centre of the true edge, and (iii) providing only a single response to a single edge. These are achieved in a four-step process [59].

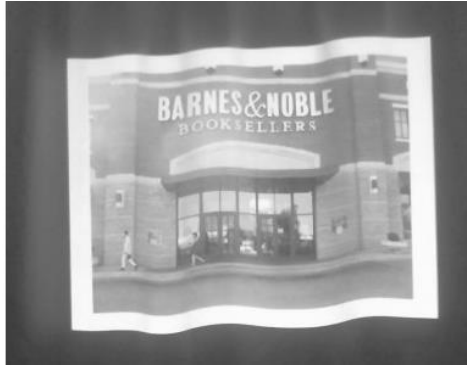
The first step involves convolving the input image with a Gaussian filter to reduce image noise.

The second step involves convolving the noise-filtered image from the first step with the Sobel kernels in Table 1 and determining the respective edge directions according to equation 2. Where gradient values in the  $u$ -direction is zero, the edge direction is taken as

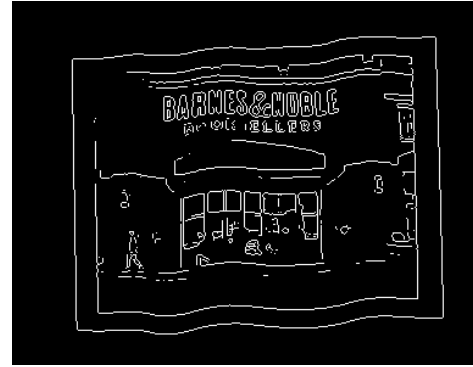
90°. Local non-maximal suppression is carried out in the third step such that all edge directions are rounded off to the nearest 45° and magnitudes of edges in all same and opposing directions are compared. Pixels with the greatest magnitudes are preserved over those with smaller gradient magnitudes. Maximal suppression results in thin edges.

The fourth step, called hysteresis thresholding [59], is carried out to remove the leftover pixels from the non-maximal suppression step brought about by noise. Here, two different thresholds – a low one and a high one are set. Pixels whose magnitudes are lower than the lower threshold are discarded. Pixels whose magnitudes are greater than the higher threshold value are kept. If pixel magnitudes fall between the two threshold values, and any of the pixel's neighbours within a 3x3 region is greater than the higher threshold, that pixel is preserved as an edge.

The performance of the various edge detectors were compared in [55], [56]. In [55] the Canny edge detector was proven to perform better than the Prewitt, Sobel and Roberts detectors as it was able to detect the edges present in a noiseless artificial image of a wheel. It also performed better in detecting the edges of a noisy natural image than the Roberts and Sobel detectors, with its performance heavily dependent on the standard deviation of the Gaussian filter and values of the lower and upper hysteresis thresholds used. In [56] the authors also concluded that the Canny edge detector performed best. The Sobel edge detector was chosen over the Canny, Roberts and Prewitt detectors in this work because of its relatively high speed of operation and it yielded satisfactory results that suited the purpose of detecting the most important features as shown in figure 10. As shown in figure 10, the most important edges to be obtained from the image of the projection for further distortion-correction processing are those forming its outer periphery.



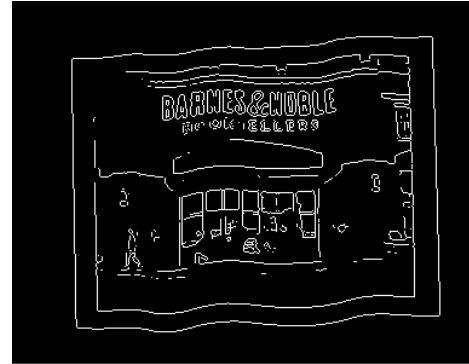
(a) Original image



(b) Prewitt edge detection (0.069s)



(c) Canny edge detection (0.132s)



(d) Sobel edge detection (0.039s)

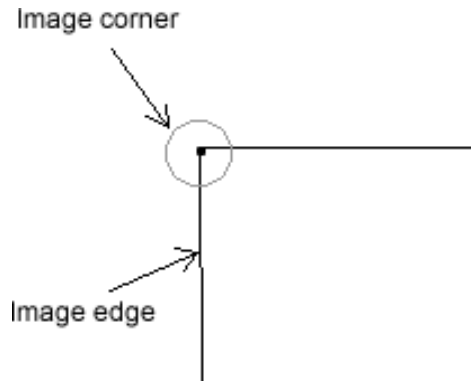


(e) Roberts edge detection (0.042s)

Figure 10: Comparison of various edge detectors to test their suitability for detecting projected image features

### 3.2.2 Image Corner Detection

An image corner is an interest point formed where two or more distinct edge segments meet [60]. Image corners of interest have been used in camera calibration [61], stereo matching [62], object tracking [63], image registration [64], stitching panoramic photographs [65] and robot navigation [66]. Figure 11 shows an example of an image corner and its constituent edges.



**Figure 11: An Image corner.**

The various image corner detectors can be divided into two classes: contour-based detectors and intensity-based detectors [67].

Contour-based corner detection is based on the amount of curvature of the respective constituent contours, edges or curves. An example is found in [68], where significant changes in the curvature of contours are described by the convolution of the first and second derivatives of a Gaussian kernel at different scales with a parameterised model of the contour segment. A typical corner is detected by either the steepness of the curve produced by the convolution of the model with the first derivative of the Gaussian or the zero-crossing of the curve derived from the convolution of the model with the second derivative of the Gaussian.

Intensity-based image corner detectors work by estimating a 'cornerness' measure for every pixel in an image to indicate the presence of a corner for each pixel. Examples of



intensity-based detectors include the Moravec method described in [69], Harris and Stephens/Plessey corner detector [70], originally described in [71], Smallest Univalued Segment Assimilation Nucleus (SUSAN) [72], Features From Accelerated Segment Test (FAST) [73], and Shi and Tomasi minimum eigenvalue method [74].

### Harris and Stephens/Plessey Corner Detector

Consider an image with pixel values  $I(u, v)$ . If an image patch of size  $n \times n$  pixels is placed over a patch of the image and shifted by amount  $(x, y)$ , the weighted sum of squared differences (SSD)  $S(u, v)$  between the two patches is given as

$$S(u, v) = \sum_x \sum_y w(x, y) (I(u+x, v+y) - I(x, y))^2 \quad (6)$$

where  $w(x, y)$  is a weighting window (eg Gaussian or rectangular) function.

Taking  $I_u$  and  $I_v$  as partial derivatives, by Taylor's expansion which eliminates  $I(x, y)$ , equation 6 can be approximated to

$$S(u, v) \approx \sum_x \sum_y w(x, y) (I_u(x, y)u + I_v(x, y)v)^2 \quad (7)$$

Equation 7 can be written in matrix form as

$$S(u, v) \approx [u \quad v] \mathbf{A} \begin{bmatrix} u \\ v \end{bmatrix} \quad (8)$$

$\mathbf{A}$  is a structure tensor called the Harris matrix [75]

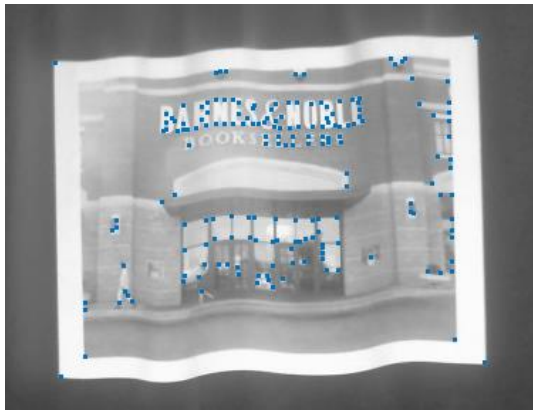
$$\mathbf{A} = \sum_x \sum_y w(x, y) \begin{bmatrix} I_u^2 & I_u I_v \\ I_u I_v & I_v^2 \end{bmatrix} \quad (9)$$

The two eigenvalues  $\lambda_1$  and  $\lambda_2$  of matrix  $\mathbf{A}$  can be found for each point and a conclusion can be derived from the relative size of the two eigenvalues:

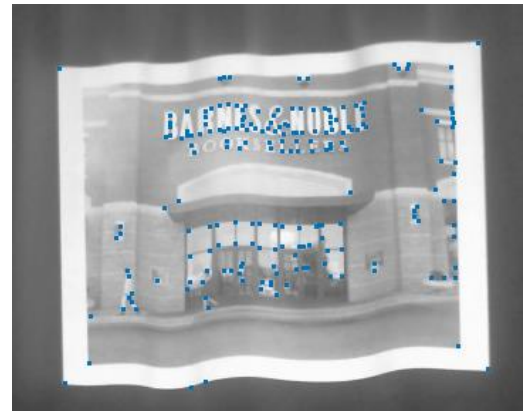
- (i) when  $\lambda_1$  and  $\lambda_2$  are both small, the point  $(u, v)$  has no feature of interest
- (ii) when  $\lambda_1$  (or  $\lambda_2$ ) is small and  $\lambda_2$  (or  $\lambda_1$ ) is large, an edge is found
- (iii) when both  $\lambda_1$  and  $\lambda_2$  are large, a corner is found.

The Harris and Stephens corner detector was used in this work to track the extreme corners of the camera image because of its relatively high speed and suitability in detecting the important corners as compared with the Shi and Tomasi minimum eigenvalue method. The results of the comparison are shown in figure 12.

The most important corners in the images of the projection needed for distortion-correction processing are the four outermost corners of the projected image.



(a) Harris and Stephens corner detection  
(0.256s)



(b) Shi and Tomasi minimum  
eigenvalue corner detection (0.319s)

**Figure 12: Comparison of the Harris and Stephens corner detector with Shi and Tomasi minimum eigenvalue corner detector in terms of detected corners and algorithm speed. Detected corners are shown as dots.**

### 3.2.3 Blob Detection

A blob (Binary Large Object) is a connected set of image pixels in an image in which some properties such as brightness or colour are approximately constant and differ in these properties to other surrounding regions in the image [76]. A typical blob detection method developed in [77] and named the Scale Invariant Feature Transform (SIFT) is said to be invariant to scale, rotation, illumination, camera viewpoint and noise. SIFT is divided into a number of steps described below:

#### (i) Scale-space extrema detection

Candidate feature points which are invariant to scale change are located using a cascaded filter approach: The input image is successively down-sampled by a factor of 2, forming octaves of the input image. Each image octave has a fixed number of Gaussian filters ( $G(u, v, \sigma)$ ) which successively differ in scale by a scale factor  $k$ . The input image in each octave is convolved with the set of Gaussian filters and the difference of the result of convolution for adjacent filter scales is obtained according to the equation:

$$D(u, v, \sigma) = (G(u, v, k\sigma) - G(u, v, \sigma)) * I(u, v) \approx (k - 1)\sigma^2 \nabla^2 G \quad (10)$$

where  $\sigma$  is the filter scale and  $\nabla^2 G$  is the Laplacian of Gaussian (LoG).

$\sigma^2 \nabla^2 G$  is a scale-normalised Gaussian and it is required for true scale invariance.

Local extrema points are respectively obtained by comparing pixels in the Difference of Gaussians (DoG) image with their immediate 8 neighbours, 9 neighbours above and 9 neighbours below in the stack of DoG images and selected if their respective values are higher or lower than those of their neighbours. These candidate feature points are also known as candidate keypoints.

#### (ii) Keypoint localisation

In this step candidate keypoints are subjected to a further test that rejects keypoints that are found to be sensitive to noise as exhibited by their low contrast and poor localisation along an edge. This test involves fitting the data surrounding each keypoint to a 3D quadratic function that is a Taylor expansion of a shifted version of  $D(u, v, \sigma)$  having its origin at the keypoint. The quadratic function is given by

$$D(\mathbf{u}) = D + \frac{\partial D^T}{\partial \mathbf{u}} \mathbf{u} + \frac{1}{2} \mathbf{u}^T \frac{\partial^2 D}{\partial \mathbf{u}^2} \mathbf{u} \quad (11)$$

#### (iii) Orientation assignment

This step involves assigning keypoints with consistent orientation based on local image properties. These values are used to form an orientation histogram for regions around each keypoint for each scale and peaks in the histogram correspond to prominent local gradient directions. The highest peak forms the orientation of the particular keypoint. New keypoints are formed if their orientation peaks are within about 80% of the highest peak. This produces multiple keypoints with the same location and scale but different orientations. A parabola is fit to the 3 histogram values closest to each peak to obtain the interpolated peak position.

#### (iv) Keypoint description

Keypoint description involves assigning a distinctive descriptor to each keypoint that is invariant to changes in illumination and 3D viewpoint. Gradient magnitudes and

orientations in neighbourhood regions around keypoints are then weighted by a keypoint scale-dependent Gaussian window. This produces 16 histograms each consisting of magnitudes in 8 orientations over  $360^\circ$  and spanning a  $16 \times 16$  region in the neighbourhood of each keypoint. Each keypoint is therefore represented by a 128-length vector with components consisting of the orientation magnitude entries of the representative histograms. The vector is then normalised to unit length to reduce the effects of changes in illumination. The SIFT blob detection was used in this work to detect common features between the projected image and the camera image of the projection as shown in figure 13.



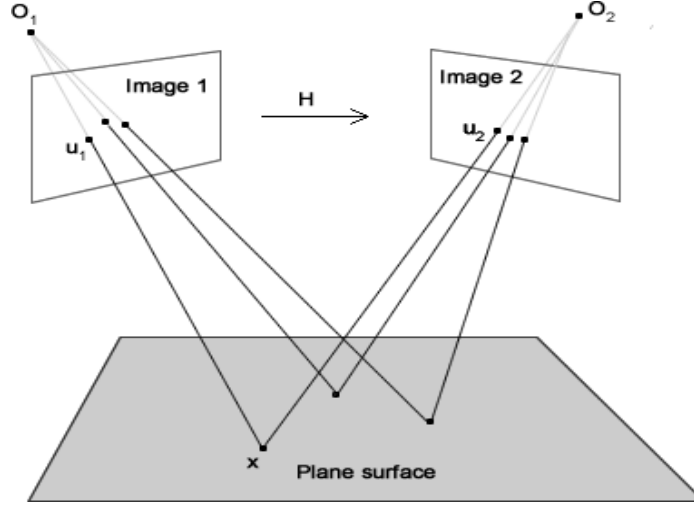
**Figure 13: Detecting common blob features between a projected image and the camera image of the projection. Common features are linked by points and coloured lines.**

### **3.3 Projections on Planar Surfaces**

#### **3.3.1 2D Homography**

Rays of light reflected from points on an object in a 2D or 3D space are seen as points in a 2D image when projected onto the planar sensor of the camera. This projection of points from a 2-or 3-D space can be described by  $3 \times 3$  or  $3 \times 4$  projective transformation matrices respectively. A  $3 \times 3$  projective transformation may be called a 2D homography. In general terms, a homography is defined as a linear mapping of points in an N-dimensional space to points in another N-dimensional space, and it is given up to an unknown scale.

A 2D homography can be a transformation of points from a planar surface to the plane of the camera sensor or a transformation of points from one image to another. The illustration is shown in Figure 14. In a homography collinear points remain collinear [78].



**Figure 14 [79]:** The projection of points  $x$  from a plane surface to respective points  $u_1$  and  $u_2$  in two image planes. 2D homographies are established between the points on the planar surface and respective points on each image plane. 2D homographies also exist between points in both images.

If a 2D point in one plane represented in homogeneous coordinate as  $(x, y, 1)^T$  is transformed by a  $3 \times 3$  homography  $H$  to a 2D point  $(u, v, 1)^T$  also in homogeneous coordinates with an unknown scale factor  $a$ , then the mapping is written as

$$a \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (12)$$

Expanding the right side of equation 12, eliminating the unknown scale factor from the left side and making  $u$  and  $v$  subjects, a pair of simultaneous equations are obtained:

$$u = \frac{h_{11}x + h_{12}y + h_{13}}{h_{31}x + h_{32}y + h_{33}} \quad (13)$$

$$v = \frac{h_{21}x + h_{22}y + h_{23}}{h_{31}x + h_{32}y + h_{33}} \quad (14)$$

Expanding equations 13 and 14, taking all the terms to the left side and converting back to matrix form, the simultaneous equations can be written as

$$\begin{bmatrix} x & y & 1 & 0 & 0 & 0 & -ux & -uy & -u \\ 0 & 0 & 0 & x & y & 1 & -vx & -vy & -v \end{bmatrix} \mathbf{h} = \mathbf{0} \quad (15)$$

where  $\mathbf{h} = [h_{11} \ h_{12} \ h_{13} \ h_{21} \ h_{22} \ h_{23} \ h_{31} \ h_{32} \ h_{33}]^T$

The desired elements of the homography  $\mathbf{H}$  can be obtained by using at least 4 pairs of point correspondences to populate equation 15 and applying the Singular Value Decomposition (SVD) of the matrix.  $\mathbf{h}$  will be the eigenvector corresponding to the smallest eigenvalue. When there are more than 4 pairs of point correspondences, the refinement of the estimates of  $\mathbf{h}$  can be done by least square error optimisation techniques.

Subgroups of a homography may be defined depending on the kind of constraints imposed on the form of the matrix  $\mathbf{H}$  [78]. These constraints give rise to geometric invariants between the original points and the transformed points. Examples of subgroups of the 2D homography include:

- (i) the affine transform where parallel lines remain parallel, and ratios of lengths on parallel lines and ratios of shape areas are constant;
- (ii) similarity transforms which preserve angles and ratio of lengths as well as all affine invariants, and
- (iii) isometric transforms which preserve lengths, angles and areas [78] [79].

The most important property of all the subgroups is that they preserve collinearity of transformed points when there are no sources of geometric distortion in the imaging system. This important property shall be used in the correction of geometric distortion through homography shaping in subsequent Chapters 4 and 5.

### 3.3.2 2D Projection onto the 2D Sensor Plane of a Camera

The transformation of points from a scene in 3D space represented in homogeneous world coordinates as  $(x, y, z, 1)^T$  onto 2D image points represented in homogeneous pixel coordinates of the camera sensor as  $(u, v, 1)^T$  is modelled by a  $3 \times 4$  camera projection

matrix  $\mathbf{P}$ . The matrix  $\mathbf{P}$  is the product of a  $3 \times 3$  intrinsic property matrix  $\mathbf{K}$  of the camera and a  $3 \times 4$  extrinsic property matrix that defines the pose (rotation and translation) of the coordinate system of the camera with respect to the reference world coordinate system.

$$a \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha & s & u_o \\ 0 & \beta & v_o \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (16)$$

In equation 16,  $\alpha$  and  $\beta$  are scaling factors along the horizontal ( $u$ ) and vertical ( $v$ ) axes of the image,  $u_o$  and  $v_o$  are pixel coordinates of the centre of the image, and  $s$  is called the skew parameter that accounts for non-square camera sensor elements. The skew parameter may be approximated to zero.

$\mathbf{R} = r_{ij}$  is the  $3 \times 3$  rotation matrix and  $\mathbf{t} = [t_x \ t_y \ t_z]^T$  the translation vector while  $a$  is an unknown scale factor and  $\mathbf{P} = [\mathbf{K}][\mathbf{R}|\mathbf{t}]$ .

When points from a plane are projected onto the camera image plane, the planar surface can be assumed to be at location  $z = 0$  of the reference world coordinate system [61].

Equation 16 therefore reduces to the form

$$a \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha & s & u_o \\ 0 & \beta & v_o \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & t_x \\ r_{21} & r_{22} & t_y \\ r_{31} & r_{32} & t_z \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \mathbf{H} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (17)$$

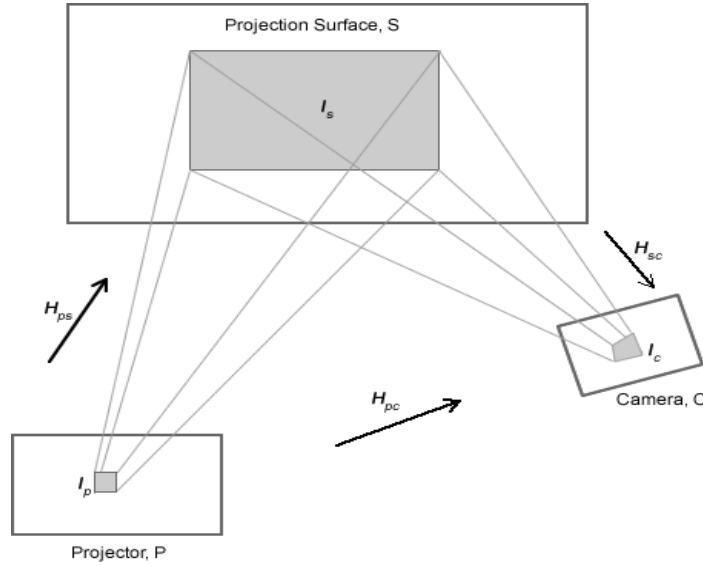
Equation 17 is also observed to be in the form of a 2D homography between the plane of projection and the sensor plane of a camera, defined up to an unknown scale  $a$ .

### 3.3.3 2D Transformations between a Projector, a Planar Projection Surface and a Camera

The model of a projector is assumed to be the same as that of the pin-hole camera but with the direction of projection reversed [80] [81]. Therefore, the projective transformation of a point from a scene in 3D space with homogeneous coordinates  $(x, y, z, 1)^T$  to its corresponding point with homogeneous coordinates  $(u, v, 1)^T$  in the 2D projected image is given by the equation

$$a \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha & 0 & u_o \\ 0 & \beta & v_o \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (18)$$

The elements of the intrinsic and extrinsic parameter matrices are identical to those defined in equation 16, with the skew parameter  $s$  assumed to be zero. The planar projection surface used for simplicity during the calibration of a projector is assumed to be located at  $z = 0$  of the reference world coordinate system [81]. Equation 18 therefore reduces to the form of equation 17 so that the 2D homography which transforms points from the projected image to points on the projection surface is defined. The respective 2D homographies between the projector, planar projection surface and a camera are shown in figure 15.



**Figure 15: Image  $I_p$  is projected from the Projector Plane  $P$  to form an image  $I_s$  on the projection surface plane  $S$  and image  $I_c$  of the projection is captured on the camera sensor plane  $C$ . 2D homographies  $H_{ps}$ ,  $H_{sc}$  and  $H_{pc}$  exist between the images formed on the three respective planes.**

Considering figure 15, the 2D transformations are defined: (i)  $H_{ps}$  transforms points  $(u_p, v_p)^T$  from the projected image to points  $(x, y)^T$  on the planar projection surface, (ii)  $H_{sc}$  defines the transformation of points  $(x, y)^T$  from the planar projection surface to points  $(u_c, v_c)^T$  on the camera image, and (iii)  $H_{pc}$  transforms points  $(u_p, v_p)^T$  from the projected image to points  $(u_c, v_c)^T$  on the camera image with the respective equations.  $a_{ps}$ ,  $a_{sc}$  and  $a_{pc}$  are the respective unknown scale factors.



$$a_{ps} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \mathbf{H}_{ps} \begin{bmatrix} u_p \\ v_p \\ 1 \end{bmatrix} \quad (19)$$

$$a_{sc} \begin{bmatrix} u_c \\ v_c \\ 1 \end{bmatrix} = \mathbf{H}_{sc} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (20)$$

$$a_{pc} \begin{bmatrix} u_c \\ v_c \\ 1 \end{bmatrix} = \mathbf{H}_{pc} \begin{bmatrix} u_p \\ v_p \\ 1 \end{bmatrix} \quad (21)$$

Combining (19), (20) and (21), we have

$$\frac{1}{a_{sc}a_{ps}} \mathbf{H}_{sc} \mathbf{H}_{ps} = \frac{1}{a_{pc}} \mathbf{H}_{pc}$$

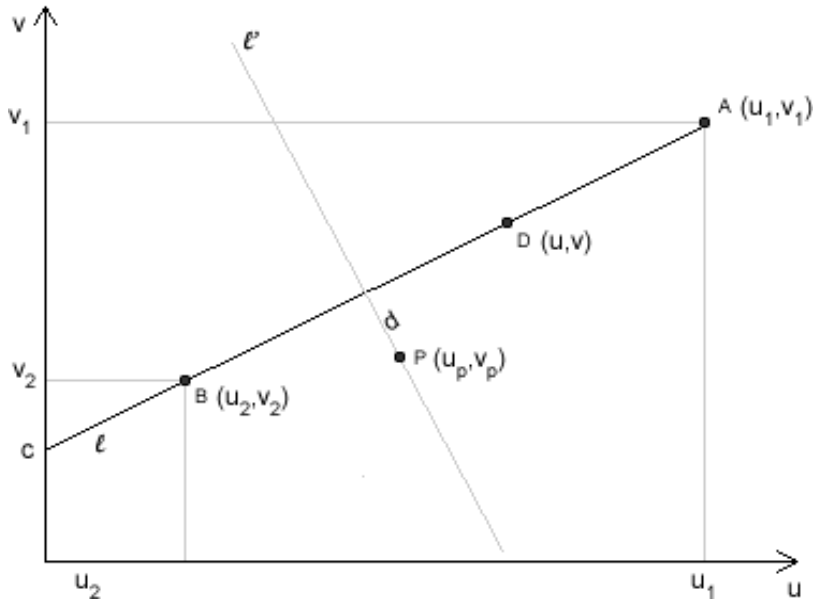
to give

$$\mathbf{H}_{pc} = \frac{a_{pc}}{a_{sc}a_{ps}} \mathbf{H}_{sc} \mathbf{H}_{ps} \quad (22)$$

Estimating the elements of the respective homographies in equations 19, 20 and 21 depends on the information available for drawing point correspondences from the images at the projector, planar projection surface and the camera and for the particular purpose. For example in [82] probes attached to a Coordinate Measuring Machine (CMM) are used to obtain point coordinates on the planar projection surface to calibrate a video projector, and in [83], a calibrated camera is used to obtain these coordinates also for projector calibration. In [84], a relationship similar to that of equation 22 between the 3 homographies is given and used for the automatic correction of keystone distortion of the projected display.

### 3.4 Straight Line Geometry

In this work, geometric distortion correction requires that straight edges of the projected image are observed as straight edges. Sampled points along straight lines from this image have to be matched with approximate corresponding points on distorted lines or edges from the camera image of the projection. The need to estimate the parameters of straight lines therefore became necessary to develop a measure of distortion for lines in acquired distorted images.



**Figure 16: A straight 2D line  $l$  described by the equation  $v = mu + c$  and points  $A$ ,  $B$  and  $D$  lying on it. Point  $P$  lies on perpendicular line  $l'$  and  $d$  is the distance from point  $P$  to line  $l$ .**

### 3.4.1 The Equation of a Straight 2D Line

The equation of a straight 2D line  $l$  of figure 16 is given in the form

$$v = mu + c \tag{23}$$

$m$  is the slope or gradient of the line and  $c$  is its intercept on the  $v$  axis.

The  $m$  and  $c$  parameters of this line can be obtained from 2 points that lie on the line.

In figure 16, suppose the points  $A$  with coordinates  $(u_1, v_1)$ , and  $B$  with coordinates  $(u_2, v_2)$ , and any other arbitrary point  $D$  with coordinates  $(u, v)$  lie on the line  $AB$ , the expression for its slope is given as

$$\frac{v_2 - v_1}{u_2 - u_1} = \frac{v_2 - v}{u_2 - u} \tag{24}$$

Rearranging equation 24 and making  $v$  its subject we have

$$v = \frac{v_1 - v_2}{u_1 - u_2} u + \frac{u_1 v_2 - u_2 v_1}{u_1 - u_2} \tag{25}$$

Comparing equations 23 and 25, slope  $m = \frac{v_1 - v_2}{u_1 - u_2}$  and intercept  $c = \frac{u_1 v_2 - u_2 v_1}{u_1 - u_2}$

In equation 25 if the line is parallel to the  $v$  axis, so that  $u_1 - u_2 = 0$ ,  $v$  will be undefined.

In such a case, the equation of the line will be expressed as

$$u = k \quad (26)$$

where  $k$  is a constant.

### 3.4.2 Estimating the Distortion of a Straight 2D Line

The distortion of a straight line can be estimated by measuring the distance of displaced points from their respective positions on the straight line. A good measure of this sort is the perpendicular distance of the displaced points from the line. The perpendicular distance of a point from a straight line is the shortest distance between the point and the line and it can be estimated from the coordinates of the point and the parameters of the line.

Referring to figure 16, the distance  $d$  of point  $P(u_p, v_p)$  to the line  $au + bv + c = 0$  is given as

$$d = \frac{|au_p + bv_p + c|}{\sqrt{(a^2 + b^2)}} \quad (27)$$

If the line is parallel to the  $v$  axis such that its equation is given as  $u = k$ , then the distance of point  $P(u_p, v_p)$  from the line will be

$$d = |k - u_p| \quad (28)$$

Sometimes it is desirable to determine the equation of the perpendicular line  $l'$  drawn from the point  $P$ . In such cases, if the gradient of the line  $l$  is given as  $m$ , then the gradient of line  $l'$  will be  $-\frac{1}{m}$ . The equation of the line  $l'$  will therefore be

$$v = -\frac{1}{m}u + \frac{u_p}{m} + v_p \quad (29)$$

From the standard equation of line  $l$  the gradient  $m = -\frac{a}{b}$ . Equation 29 will become

$$v = \frac{b}{a}u - \frac{bu_p}{a} + v_p \quad \text{or in standard form,}$$

$$bu - av - bu_p + av_p = 0 \quad (30)$$

If line  $l$  is parallel to the  $u$  axis its gradient would be zero, line  $l'$  would be parallel to the  $v$  axis and its gradient would be undefined. In such a case, the equation of  $l'$  would simply be

$$u = u_p \quad (31)$$

The equations of lines  $l$  and  $l'$  can be solved simultaneously to obtain the point at which they both intersect.

### 3.5 Image Warping Techniques

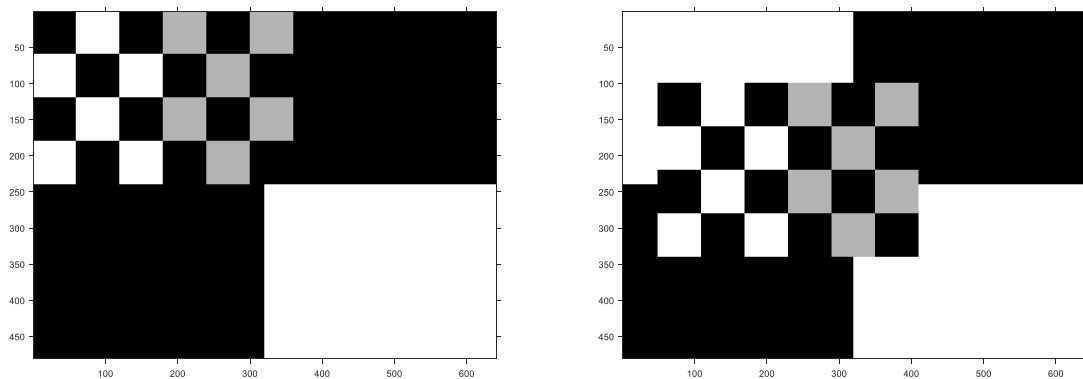
Several methods of warping an image were discussed in Section 2.4. This section presents their parametric equations and respective images to which they were applied. In all the examples given, an image  $I$  with coordinates  $(x, y)$  is warped to image  $I'$  with new coordinates  $(u, v)$  by a warping function  $f(x, y)$ . The following transformations were done using normalised image coordinates in the interval  $(-1, 1)$ .

#### 3.5.1 Translation

Translation of image coordinates either along rows or columns is given as:

$$u = x + a, \quad v = y + b \quad (32)$$

$a$  and  $b$  are respective constant pixel displacements along the columns and rows of the image.



(a) Original image

(b) Transformed image

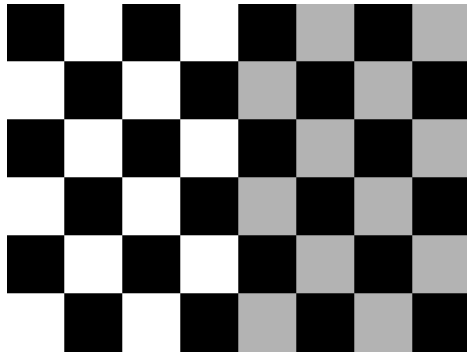
**Figure 17: Translation of a region of pixels in an image. The pixels in the checker board pattern in the upper left region of the image in (a) are translated by  $a = 50$  and  $b = 100$  pixels horizontally and vertically respectively to produce the image in (b).**

### 3.5.2 Procrustes Transformation

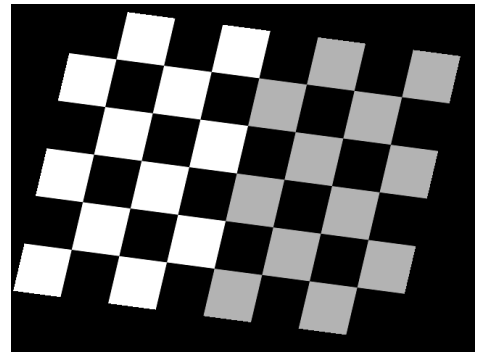
The Procrustes transformation involves uniform magnification, rotation, and scaling of the original image [35]. It is given as

$$u = cx \cos \theta + cy \sin \theta + a, \quad v = -cx \sin \theta + cy \cos \theta + b \quad (33)$$

where  $a$  and  $b$  are constant translation parameters,  $c$  is a constant scaling parameter and  $\theta$  represents angular rotation of image coordinates in degrees.



(a) Original image



(b) Procrustes-transformed image

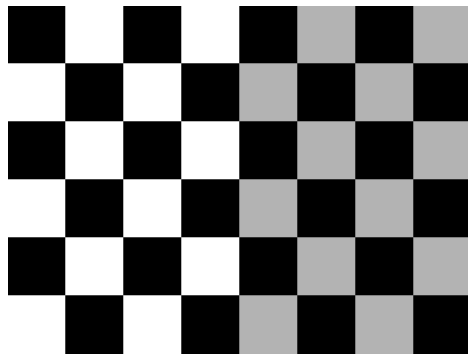
Figure 18: Procrustes transformation of an image where  $a = 20$ ,  $b = 10$ ,  $c = 1.2$ ,  $\theta = 10^\circ$ .

### 3.5.3 Affine Transformation

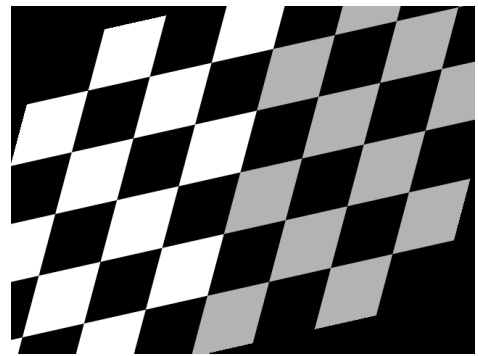
The affine transformation of an image is the generalisation of the Procrustes transformation and is given as

$$u = a_1x + a_2y + a_3, \quad v = b_1x + b_2y + b_3 \quad (34)$$

The affine transformation permits different degrees of stretching and shearing along the rows and columns of an image [35].



(a) Original image



(b) Affine-transformed image

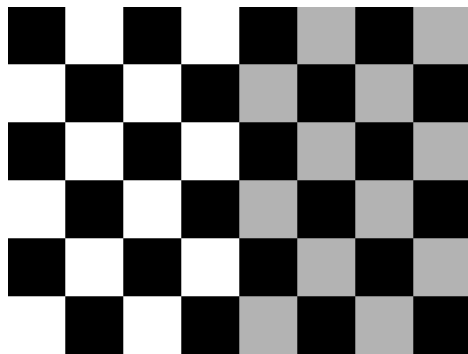
Figure 19: Affine transformation of an image where  $a_1 = 1, a_2 = 0.2, a_3 = 10$  and  $b_1 = 0.3, b_2 = 1, b_3 = 20$ .

### 3.5.4 Perspective Transformation

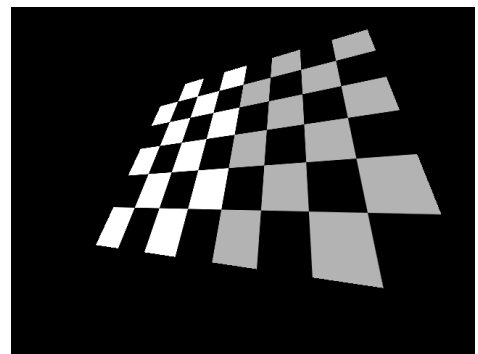
The perspective transformation of an image has the form

$$u = \frac{a_1x + a_2y + a_3}{c_1x + c_2y + 1}, \quad v = \frac{b_1x + b_2y + b_3}{c_1x + c_2y + 1} \quad (35)$$

It generalises an affine transformation [79]. It maps straight lines to straight lines and preserves conic sections [35].



(a) Original image



(b) Perspective-transformed image

Figure 20: Projective-transformed image with  $a_1 = 1.5, a_2 = 0.2, a_3 = 0.1; b_1 = 0.3, b_2 = 1, b_3 = 0.4; c_1 = 0.5, c_2 = 0.5, c_3 = 0.9$

### 3.6 Image Warping with Radial Basis Function Interpolation

A Radial Basis function (RBF)  $\phi$  is defined as a real-valued function of the distance of a point to the origin, or the distance of the point from some other point. An RBF satisfies the property  $\phi(x) = \phi(\|x - c\|)$  where  $\|x - c\|$  defines the distance of the point  $x$  from the reference point  $c$  [85]. Radial basis functions are used to interpolate a continuous function whose value is known only at a set of discrete points at other points where the value of the continuous function is unknown. An RBF interpolant is a linear combination of translates of a Radial Basis Function  $\phi(\|\cdot\|)$  with a low-degree polynomial [45]. According to the explanation given in [47], a deformable structure at rest will have control points  $P$  with coordinates  $\mathbf{x}_k^0$ . When the structure is deformed, the control points will have coordinates  $\mathbf{x}_k$ . An interpolating function  $f$  is needed to approximate the coordinates  $\mathbf{x}$  of other points on the deformed structure that also have coordinates  $\mathbf{x}^0$  when the structure is at rest so that

$$\mathbf{x} = f(\mathbf{x}^0, \mathbf{x}_k^0, \mathbf{x}_k) \quad (36)$$

Warping an image with an interpolating function  $f$  involves calculating the approximate coordinates of points in the image from the coordinates of the control points with known displacement. According to [47] the interpolating function  $f$  should satisfy the following conditions:

- (i) Translation and rotational invariance.
- (ii) The interpolation condition given in equation 36 must be satisfied for all control points.
- (iii) The displacement of any point should depend linearly on the displacement of the control points.

The realisation of interpolation function  $f$  that also satisfies the conditions (i) to (iii) above can be given by a linear combination of  $n$  translates of an RBF with a polynomial to give

$$s(\mathbf{x}) = \sum_{i=1}^n \lambda_i \phi(\|\mathbf{x} - \mathbf{x}_i^k\|) + \sum_{j=1}^m \beta_j g(\mathbf{x}) \quad (37)$$

where  $\lambda_i \in \mathbb{R}$  are coefficients of the RBF translates  $\phi$ ,  $g(\mathbf{x})$  is a low degree polynomial with coefficients  $\beta_j$ , and  $n$  is the number of control points used. The coefficients are chosen so that for all polynomials  $q$  of degree less than or equal to that of  $g$ , the following condition must be satisfied [45], [86]:

$$\sum_{j=1}^n \lambda_j q(\mathbf{x}) = 0 \quad (38)$$

The equations 37 and 38 can be combined into matrix form to yield

$$\begin{bmatrix} \mathbf{A} & \mathbf{P} \\ \mathbf{P}^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{\lambda} \\ \boldsymbol{\beta} \end{bmatrix} = \begin{bmatrix} \mathbf{h} \\ \mathbf{0} \end{bmatrix} \quad (39)$$

where  $\mathbf{A}$  is an  $n \times n$  matrix with elements  $A_{ij} = \phi(\|x_{k_i} - x_{k_j}\|)$  (40)

$\mathbf{P}$  is a matrix defined by the constraint in equation 38 for first-degree polynomials.

$$\mathbf{P} = \begin{bmatrix} 1 & x_{k_1}^0 \\ 1 & x_{k_2}^0 \\ 1 & x_{k_3}^0 \\ \vdots & \vdots \\ 1 & x_{k_n}^0 \end{bmatrix} \quad (41)$$

When matrix  $\mathbf{A}$  is invertible, the  $\boldsymbol{\lambda}$  and  $\boldsymbol{\beta}$  coefficients are obtained by solving [47]

$$\boldsymbol{\lambda} = \mathbf{A}^{-1} \mathbf{v} - \mathbf{A}^{-1} \mathbf{P} \mathbf{A}_p \mathbf{P}^T \mathbf{A}^{-1} \mathbf{h} \quad (42)$$

$$\boldsymbol{\beta} = \mathbf{A}_p \mathbf{P}^T \mathbf{A}^{-1} \mathbf{h} \quad (43)$$

$$\text{where } \mathbf{A}_p = (\mathbf{P}^T \mathbf{A}^{-1} \mathbf{P})^{-1} \quad (44)$$

For a 2D image, Equations 42 and 43 are solved to obtain the set of coefficients  $\boldsymbol{\lambda}^x, \boldsymbol{\lambda}^y$  and  $\boldsymbol{\beta}^x, \boldsymbol{\beta}^y$  from respective displacement of points in each of the  $x$  and  $y$  dimensions of the image. Equation 37 can be rewritten to reflect the new coordinates of points in a warped 2D image as

$$s_x(\mathbf{x}) = \sum_{i=1}^n \lambda_i^x \phi(\|\mathbf{x} - \mathbf{x}_{k_i}\|) + \beta_1^x + \beta_2^x x + \beta_3^x y \quad (45)$$

$$s_y(\mathbf{x}) = \sum_{i=1}^n \lambda_i^y \phi(\|\mathbf{x} - \mathbf{x}_{k_i}\|) + \beta_1^y + \beta_2^y x + \beta_3^y y \quad (46)$$

There are various RBFs in literature that can take the place of  $\phi$  in the interpolating equations above, and a number of them were compared for efficiency in [86]. Four of them are listed in Table 2. The test to determine the effectiveness of the first three RBFs in Table 2 in pre-warping a projected image to correct geometric distortions in projected displays is carried out in Section 4.3. Results are presented in Section 4.4.



**Table 2: Examples of Radial basis functions [86]**

No.	Name	$\phi(x)$
(i)	Thin plate spline	$x^2 \log(x)$
(ii)	Multiquadric Biharmonics	$\sqrt{a^2 + x^2}$
(iii)	Inverse Multiquadric Biharmonics	$\sqrt{\frac{1}{a^2 + x^2}}$
(iv)	Gaussian	$e^{-ax^2}$

The parameter  $a$  in (ii) to (iii) of Table 2 is a scaling factor which controls the shape of the RBF. The value of  $a$  also affects the coordinates of the warped pixels. By using normalised pixel values in the range (-1,1) in both image directions, the value of  $a$  is determined experimentally and set to 0.01 to ensure that all warped pixels stayed within the specified range.

### 3.7 Image Similarity Measurement

After correcting the geometric distortion of a projected image, there is need to objectively measure the effectiveness of the correction system. A typical way of doing this is to compare the image of the geometrically-corrected projection with a desired standard derived from the homography-transformed projected image. There are several ways to measure the similarity between two images and two of them are presented below.

#### 3.7.1 Normalised Cross-correlation Coefficient (NCC)

Normalised cross-correlation is a commonly used metric in evaluating the degree of similarity or dissimilarity of a set of images being compared. It is less sensitive to changes in illumination in the images being compared than the ordinary cross correlation [87]. According to [88], let  $f(x, y)$  be the intensity value of the image  $f$  of size  $M_x \times M_y$  at any point. Let template  $t$  of size  $N_x \times N_y$  represent a pattern whose position in image  $f$  is to be determined. The NCC value  $\gamma$  at each point  $(u, v)$  of  $f$  is given by the expression

$$\gamma = \frac{\sum_{x,y} (f(x,y) - \bar{f}_{u,v})(t(x-u, y-v) - \bar{t})}{\sqrt{\sum_{x,y} (f(x,y) - \bar{f}_{u,v})^2 \sum_{x,y} (t(x-u, y-v) - \bar{t})^2}} \quad (47)$$

$\bar{f}_{u,v}$  is the mean value of  $f(x,y)$  calculated over the area of the template  $t$  shifted to  $(u,v)$  and given by the expression

$$\bar{f}_{u,v} = \frac{1}{N_x N_y} \sum_{x=u}^{u+N_x-1} \sum_{y=v}^{v+N_y-1} f(x,y) \quad (48)$$

$\bar{t}$  is the mean value of template  $t$  given by the expression

$$\bar{t} = \frac{1}{N_x N_y} \sum_{x=u}^{u+N_x-1} \sum_{y=v}^{v+N_y-1} t \quad (49)$$

The value of  $\gamma$  ranges between -1 and 1. When the NCC is used as a measure of the effectiveness of the geometric distortion process, the value of  $\gamma$  for the geometrically-corrected template should be closer to 1 than that of the geometrically-distorted template.

### 3.7.2 Phase Correlation

Phase correlation is an example of a Fourier-based image similarity measurement method. It computes the cross-power spectrum of a reference image and a template, and looks for the peak in its inverse. It is based on the Fourier shift theorem that states that a shift in the coordinate frame of two functions represents itself in the Fourier domain as linear phase differences between the functions [89]. It is a good choice where images are corrupted by frequency-dependent noise [49]. If  $f_1(x,y)$  and  $f_2(x,y) = f_1(x-x_o, y-y_o)$  are two images and  $\hat{f}_1(u,v)$  and  $\hat{f}_2(u,v)$  are their respective Fourier transforms, then according to Fourier shift property,

$$\hat{f}_2(u,v) = \hat{f}_1(u,v) e^{-i(ux_o + vy_o)}$$

The normalised cross power spectrum for both images is given as

$$\frac{\hat{f}_2(u,v) \hat{f}_1(u,v)^*}{|\hat{f}_1(u,v) \hat{f}_1(u,v)^*|} = e^{-i(ux_o + vy_o)} \quad (50)$$

where \* indicates complex conjugate.

### **3.8 Chapter Summary**

The main focus of this chapter has been to present the theories underlying the image processing techniques used in this work. It included the following discussions:

(i) A discussion of image feature detection techniques including edge, corner and blob detection techniques. Detected image features are used in drawing up correspondences between geometrically transformed versions of an image.

(ii) Projections on planar surfaces including the 2D homography and its use in modelling the geometrical transformations between image points on the projection plane of the projector, the planar projection surface, and the sensor plane of the camera.

(iii) The geometry of the straight line including the derivation of its equation and that of the distance of a point from the line to be used as a measure of the level of distortion of the straight line.

(iv) Illustrative examples of parametric image warping techniques discussed in Section 2.4.

(v) The derivation of the RBF coefficients used to estimate the warping of points (pixels) in an entire image from displacements of a set of control points in the image.

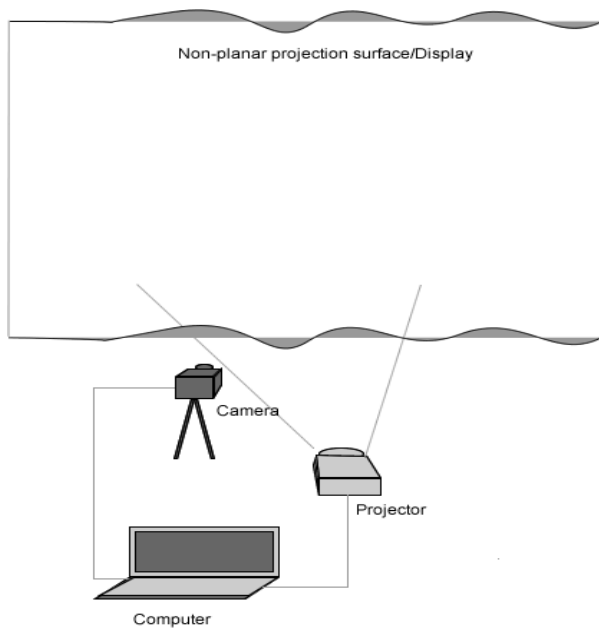
(vi) The presentation of the Normalised Correlation Coefficient (NCC) and Phase Correlation as respective spatial and frequency domain techniques used as measures of similarity between transformed versions of an image.

# CHAPTER FOUR

## Geometric Distortion Correction for Static Projection Displays

### 4.1 Introduction

This chapter presents the techniques developed in this work to correct geometric distortions when non-distorted images are projected onto static non-planar displays or surfaces. It begins in Section 4.2 by presenting a simple illustration of the projection of a straight line onto a quadric surface, the form of distortion of the line on the surface, and the derivation of the kind of warp to be applied to the line in order to obtain its undistorted view on the quadric surface. The methods used to measure and correct geometric distortion of projections on static non-planar surfaces using a projector-camera system are presented in Section 4.3 through Section 4.6. While one method uses a calibration image the other does not. The tests carried out to determine the effectiveness of each method in correcting geometric distortion of images projected onto static non-planar surfaces are described and results of these tests are presented in the subsequent sections. The static projection system used in the tests consist of the following: (i) An Epson EMP-S3 projector mounted on a fixed platform, (ii) A white window curtain used as a non-planar projection surface, with the top end attached to and hanging from a string and the bottom end left free, (iii) A Logitech C615 webcam with a  $640 \times 480$  (VGA) pixel resolution held in place on a rigid support, (iv) A computer with a Windows 10 Operating System and an Intel Celeron N2830 2.16GHz processor. A GUI was developed in Matlab to control the projection, acquisition and processing of images respectively from the projector and camera connected to the computer through VGA and USB 2.0 ports respectively. The GUI is shown in appendix A3. The system setup is shown in figure 21.



**Figure 21: A projection system consisting of a projector, camera, projection surface and computer**

## **4.2 Illustrating Geometric Distortion of a Straight Line Projected onto a Non-planar Surface**

### **4.2.1 Pictorial Illustration**

This section illustrates how geometric distortion occurs when a straight line is projected onto a non-planar surface and how the geometric distortion created can be corrected.

A simple illustration in figure 22 shows the profile of the projection of a straight line on a quadric surface. The illustration using a cylindrical surface can be generalised for other quadric and non-parametric non-planar surfaces. The illustration is described as follows:

(i) The line  $l$  in the figure represents a horizontal line in an image being projected onto the surface  $S$  of a cylinder, and only the profile of the orthographic projection of section  $AB$  of the line is seen according the viewpoint illustrated.

(ii) The cylinder stands upright so that its base is parallel to the  $xy$  plane. The projector is assumed to be tilted at an acute angle  $\alpha$  to the horizontal  $xy$  plane.

(iii) Rays of light forming a plane emanating from  $l$  strike  $S$  to form an elliptical curve  $C_1$  on  $S$ .  $C_1$  represents the distorted profile or image of the line that needs to be corrected so that what should be observed on  $S$  is circle  $C_2$ .

(iv) The light rays emanating from the back-projection of  $C_2$  from  $S$  to the image plane  $P$  of the projector form section  $A'B'$  of arc  $l'$  on  $P$ . These rays are actually sections of a collection of horizontally-stacked parallel planes, all forming acute angle  $\alpha$  with the  $xy$  plane. The shape of  $l'$  in this case is seen to be opposite to that of  $C_1$ .

(v) The back-projection process described in (iii) and (iv) may be reversed in order to observe an undistorted projection of line  $l$ . For this to take place, line  $l$  should first be warped to arc  $l'$  and then projected.

The illustration in figure 22 is an orthographic representation but in reality the projector-surface-camera system is a perspective projective system. The principles would also apply in actual systems as projective transformations also transform quadric curves to quadric curves. An algebraic explanation and solution to the geometric distortion problem caused by an inclined projection of a line onto a non-planar surface is given in the following section 4.2.2.

#### 4.2.2 Algebraic Representation of a Plane and a Cylinder

The equation of a cylinder of radius  $r$  as seen in figure 22 is given as

$$x^2 + y^2 = r^2 \quad (51)$$

$$\text{where } x = r \cos \theta, \quad y = r \sin \theta \quad (52)$$

$\theta$  is the angle of projection of a point on the surface of the cylinder onto the  $x$  axis.

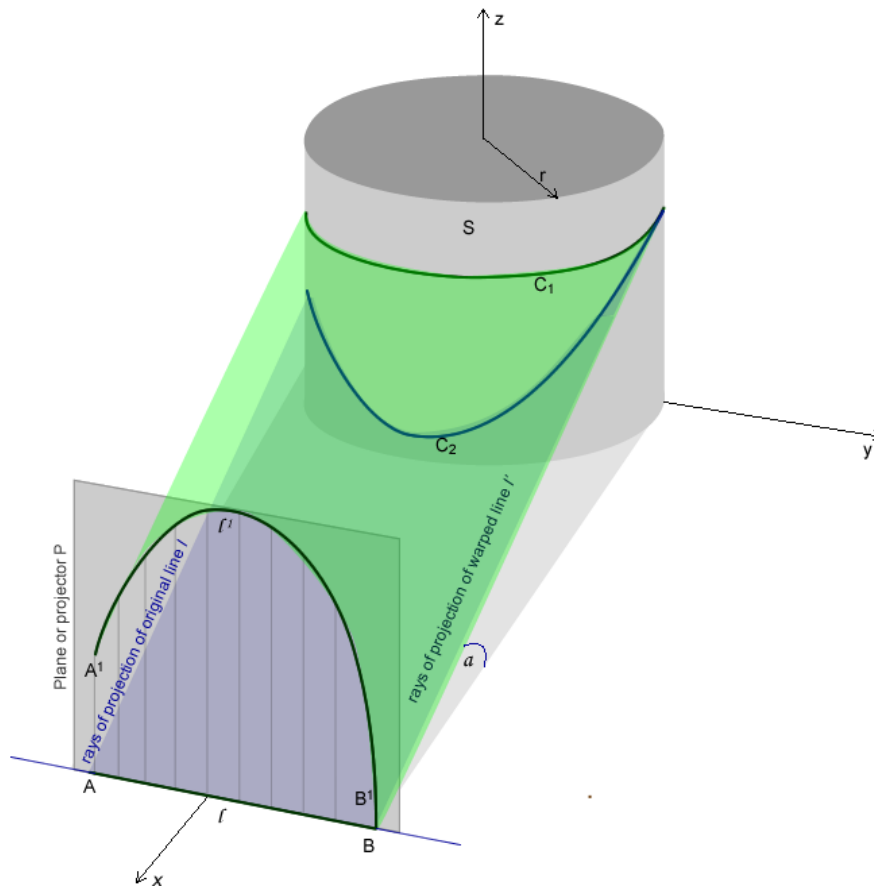
The general equation of a plane is given as

$$ax + by + cz = d \quad (53)$$

To obtain the points of intersection of a plane with a cylinder, the simultaneous equations 51 to 53 are solved. Substituting for  $x$  and  $y$  in equation 53 and solving for  $z$  gives

$$z = \frac{d - r(a \cos \theta + b \sin \theta)}{c} \quad (54)$$

Equation 54 shows that the  $z$  coordinates of the points of intersection of the plane with the surface of the cylinder vary with the value of  $\theta$  as the other parameters  $a, b, c, d$ , and  $r$  are constant. With reference to figure 22, it is assumed that the range of  $\theta$  on the visible surface of the cylinder is  $-\frac{\pi}{2} \leq \theta \leq \frac{\pi}{2}$ .



**Figure 22: Geometric illustration of the projection of a 2D image line onto a 3D or non-planar surface  $S$ .  $C_1$  represents the distorted profile of the line  $l$  projected onto quadric surface  $S$ .  $C_2$  is the corrected profile of the line after pre-warping it from  $l$  to  $l'$ .**

### The pose of a projector

The effect of the pose of a projector with respect to a quadric surface can be observed by tracing the plane of light rays being projected from a straight horizontal line onto the surface of the cylinder. The amount of inclination of the projector determines the amount of inclination of the plane of light rays emanating from the projected line. The equation of this plane containing the light rays can be derived from the coordinates of a point on the plane and a vector specifying a line that is perpendicular to the plane [90], and both can be derived from the angles of inclination of the plane to the respective coordinate axes. The equation of a plane containing the rays of light from the projected line that is inclined to the  $x$  and  $y$  axes by angles  $\beta$  and  $\alpha$  respectively can be given as

$$\sin(\alpha) x - \cos(\alpha) \sin(\beta) y + \cos(\alpha) \cos(\beta) z = 0 \quad (55)$$

Comparing equations 53 and 55,  $a = \sin \alpha$ ,  $b = -\cos \alpha \sin \beta$ ,  $c = \cos \alpha \cos \beta$ ,  $d = 0$ .

These can be substituted into equation 54 to obtain the profile of the projection on the surface of the cylinder. Figure 23 shows the plotted profiles for different values of  $\alpha$  and  $\beta$ .

### Deriving the warp to be applied to the projected line

The warping of line  $l$  to correct the observed geometric distortion for this system can be derived as follows:

From the illustration in figure 22, it is desired that the  $z$  coordinate of curve  $C_1$  at  $\theta_1$  equals the  $z$  coordinate of the curve at another angle  $\theta_2$ . Equation 54 becomes

$$\frac{d_1 - r(a \cos \theta_1 + b \sin \theta_1)}{c} = \frac{d_2 - r(a \cos \theta_2 + b \sin \theta_2)}{c} \quad (56)$$

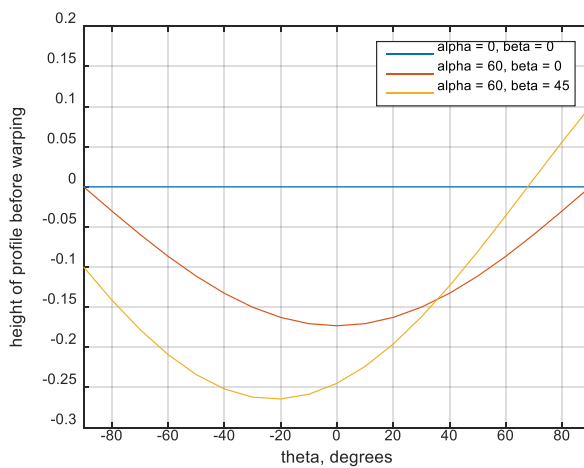
rearranging and solving for  $d_2$ ,

$$d_2 = d_1 - ar(\cos \theta_1 - \cos \theta_2) - br(\sin \theta_1 - \sin \theta_2) \quad (57)$$

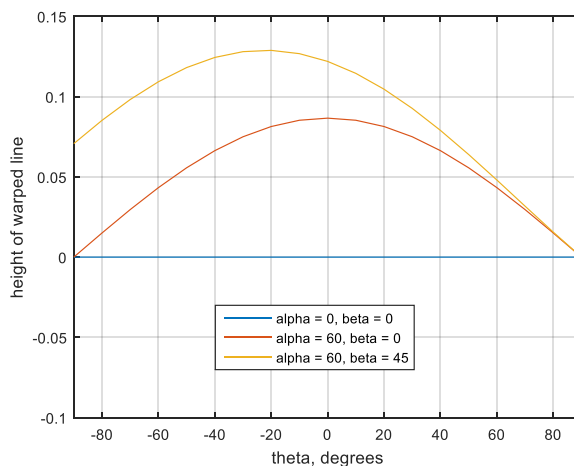
Equation 57 gives an expression for the translation of plane  $d_1 = ax + by + cz$  to  $d_2 = ax + by + cz$ . By making  $\theta_1$  the target and varying  $\theta_2$ , a set of new planes can be derived. Note that all the planes in the set are parallel.



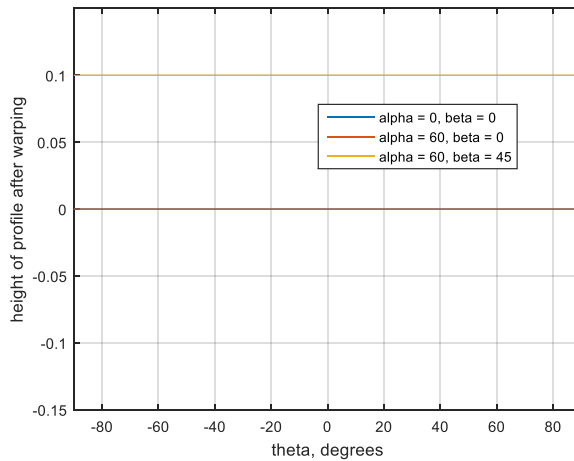
In conclusion, the resulting warped line is the locus of sections of the set of parallel planes defined by angle  $\theta$ , and obtaining  $d_2$  for the respective planes is tantamount to deriving the shape of the warped line. Figure 23(b) shows the respective shapes of the warped lines as obtained by evaluating  $d_2$  from equation 57, and making the values of  $z$  at other angles  $\theta_2$  equal to  $z$  at  $\theta_1 = \frac{\pi}{2}$  ( $90^\circ$ ). The respective values of  $d_2$  thus obtained can be substituted back into equation 54 to obtain the desired profile of the projection of the warped line on the quadric surface. Figure 23(c) shows the respective profiles of the projected warped lines on the surface of the quadric.



**(a) The profile curves for various inclinations of the plane of rays to the horizontal  $x$  and  $y$  axes**



**(b) the derived warped shapes of the horizontal line required to change the respective observed profiles on the quadric surface**



(c) the derived respective projection profiles after projecting the warped lines in (b)

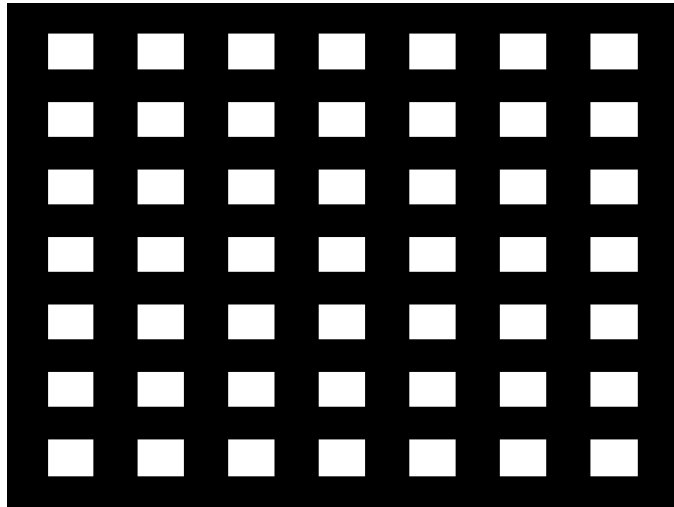
**Figure 23: Plots of the profile curve of a horizontal line projected onto a quadric (cylindrical) surface.**

### 4.3 Geometric Distortion Measurement of Image Lines and Geometric Distortion Correction using a Calibration Image

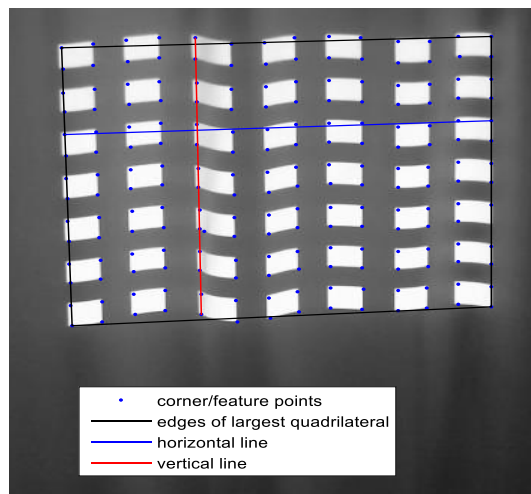
#### 4.3.1 The Calibration Image and Measure of Distortion of its Grid Lines

The calibration image shown in figure 24 was used to estimate the geometric distortion suffered by straight image lines projected on the non-planar projection surface. It consisted of white rectangles arranged in equal number of rows and columns against a uniform black background. The corners of the rectangles forming rows and columns of horizontal and vertical lines respectively represented distinctive image features to be used for the estimation of distortion of straight projected image lines caused by the shape of the projection surface. Corresponding corner points were obtained from the camera-captured image of the projection of the calibration image on the non-planar surface. A series of horizontal and vertical lines were formed by linking the two points at the end of each row and column of feature points formed as shown in figure 25. The parameters of each line was estimated and its distortion measured by calculating the deviation of each member point from the line and finding the average as described in Section 3.4.2. The plot of the average deviation of points from each line of the camera-captured projection of the calibration image is shown in figure 25. It is used as an indication of the amount of distortion that each line of the calibration image suffered as the calibration image was projected onto

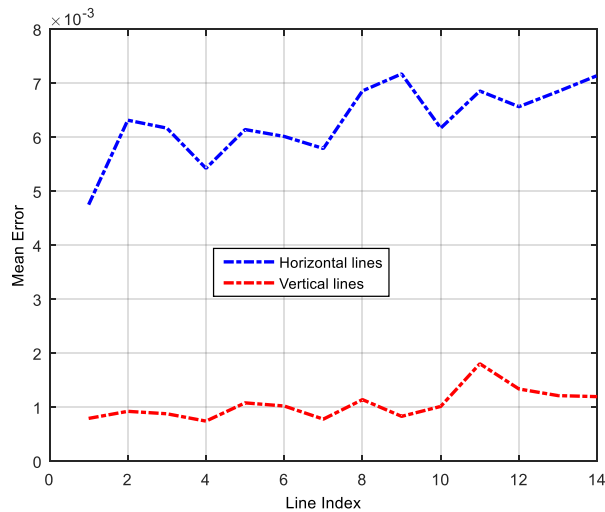
the non-planar surface. The approximate minimum and maximum average deviation values in pixels in both image directions obtained for the 640 x 480 pixel image of the calibration image shown in figure 25 and plotted in figure 26 are compared in Table 3. The figures and table all reveal that projected horizontal lines of the calibration image with larger average deviation values suffer greater distortion on the projection surface than the vertical lines of the calibration image. Note that because both types of image lines slant their deviation values are represented in horizontal and vertical pixel components.



**Figure 24: Calibration image consisting of  $14 \times 14$  feature points projected onto the non-planar surface to measure and correct the distortion of the projected image**



**Figure 25: Camera capture of the calibration image projected onto the non-planar projection surface showing the detected corner points, horizontal and vertical lines formed by linking two extreme corner points, and the quadrilateral formed by linking the corner points at the four extreme corners of the image.**



**Figure 26: A plot of average deviation of member feature points from the horizontal and vertical lines shown in figure 25. Normalised pixel values in the range (-1,1) were used**

**Table 3: Comparing approximate minimum and maximum deviations in 2 image directions for horizontal and vertical lines of the projected calibration image**

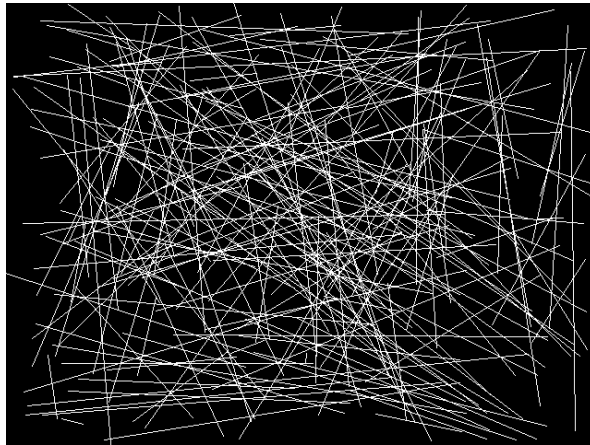
	Horizontal (u) and vertical (v) image directions	
	Min deviation (px)	Max deviation (px)
Horizontal image line	1.06	1.64
Vertical image line	0.10	0.27

### 4.3.2 Measure of Distortion/Straightness of Straight Lines in an Image

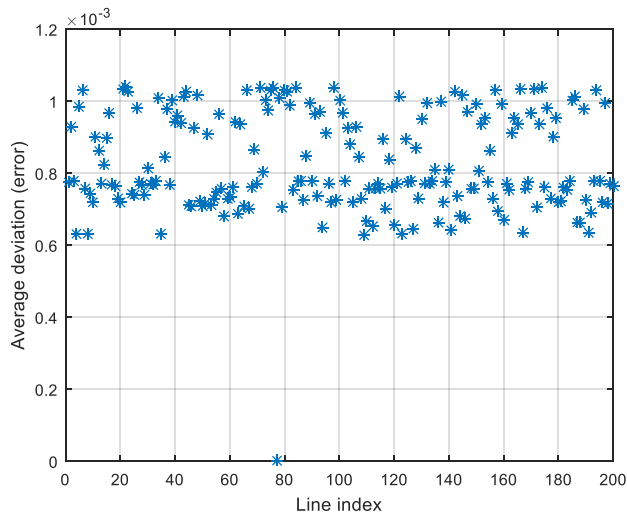
Lines observed to be perfectly straight in an image also have geometric distortion errors associated with them as their member points may have very small non-zero deviation values from the lines. This deviation is caused by quantisation errors in representing the constituent points of the lines as discrete image pixels.

The test carried out to estimate the level of distortion of the straight lines in an image involved the following:

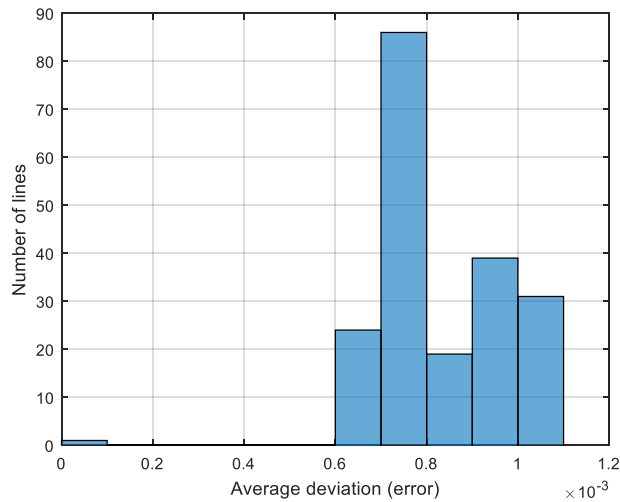
(i) Randomly choosing two distinct points with absolute and normalised pixel coordinates, and finding the parameters of the line joining them as described in Section 3.4.1, (ii) Using the equation of the line to determine the absolute pixel coordinates of all the points in a  $640 \times 480$  pixel-sized image that belong to the line, (iii) Converting the absolute pixel coordinates of the member points found to normalised coordinates and calculating the deviation of each point from the line using equation 27, and (iv) calculating the average deviation of the points from the line to use as a measure of distortion for each random image line drawn. The plot of average deviation for 200 randomly-generated image lines in figure 27 is shown in figure 28. The average deviation plot shows that the straightest lines in an image with normalised pixel values in the interval  $(-1,1)$  can have average deviation values ranging from 0 for perfectly horizontal and vertical lines to about 0.0011 for slanted lines. The distribution of average deviations is also shown as a histogram in figure 29.



**Figure 27: 200 Randomly-generated image lines**



**Figure 28: Plot of average deviations for lines shown in figure 27**

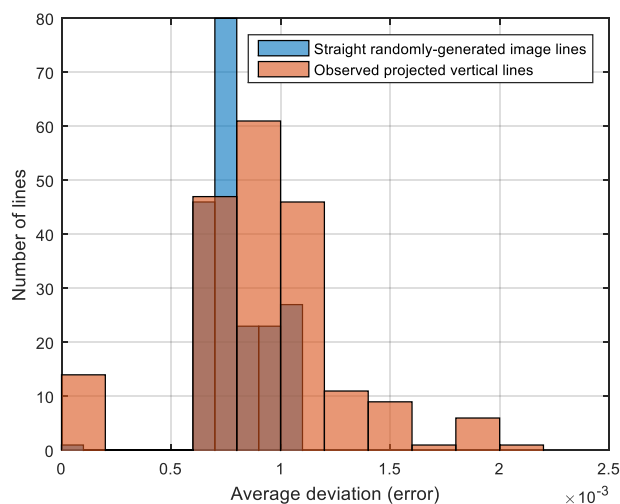


**Figure 29: Histogram showing the distribution of average deviation values for the 200 randomly-generated lines in figure 27.**

### 4.3.3 Measuring the Distortion of Projected Horizontal and Vertical Lines of the Calibration Image

When the calibration image projected onto the non-planar surface was captured by the camera, it was observed that the horizontal lines formed by the feature points on the image appeared to be more distorted than the vertical ones. This was due to the crests and troughs on the projection surface lying largely in the horizontal direction as the surface hung freely and naturally from a horizontal string. In the presence of these crests and troughs, the

surface normals from feature points of the calibration image projected onto the non-planar surface are largely parallel in the case of vertical lines and non-parallel in the case of the horizontal lines. The respective distortions of horizontal and vertical lines in the camera image of the projection were compared with the distortions of the standard straight image lines described in Section 4.3.2. Average deviation values were obtained from 196 horizontal and vertical lines by projecting the calibration image onto the surface, capturing the projection with the camera and calculating the distortion of each line for different poses of the camera and projector and shapes of the projection surface. The calibration image used consisted of 14 horizontal and vertical lines respectively. The distribution histograms of deviations of member points of the projected horizontal and vertical lines of the calibration image and straight image lines are shown in figures 30, 31 and 32 to illustrate the similarities and differences between them.

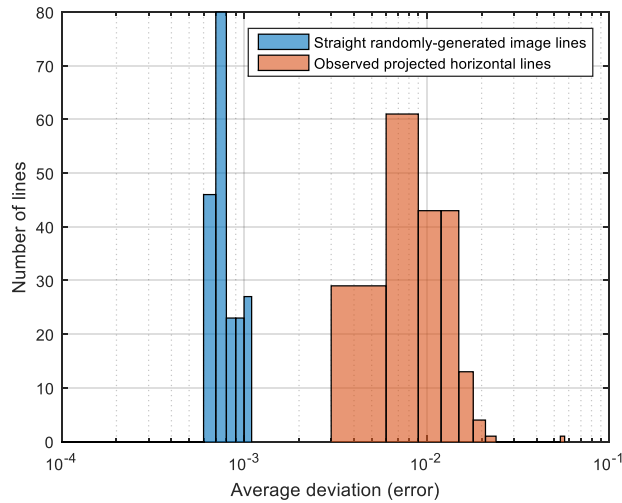


**Figure 30: Histograms of distributions of average deviations of points from the projected vertical lines of the calibration image (brown) and average deviations of points from 200 randomly-generated straight image lines (blue).**

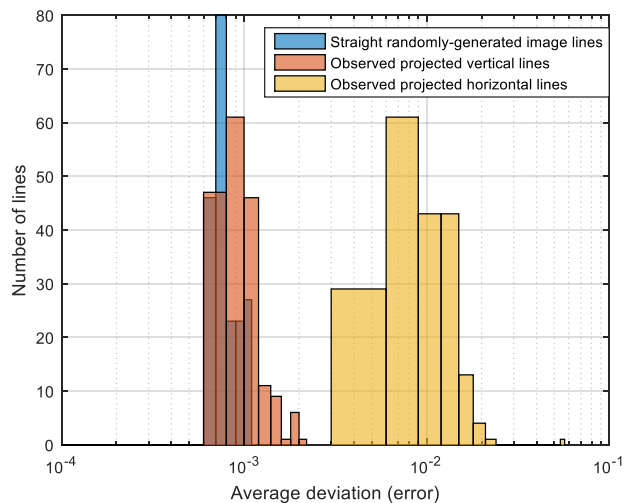
The histogram in figure 30 shows that the distribution of average deviation for projected vertical lines of the calibration image has values in the region of that of straight image lines. Therefore for simplicity during the geometric distortion correction process, the distortion of vertical lines may be ignored. The distribution of deviation errors of points from the horizontal lines of the projected calibration image in figures 31 and 32 shows much greater average deviation values than those of vertical lines of the calibration image and randomly-



generated lines respectively, and a clear separation of the two distributions. The projected horizontal lines are significantly more distorted than the other sets of lines. The geometric distortion of horizontal lines of the grid would therefore require correcting.



**Figure 31: Histograms of distributions of average deviations of points from the projected horizontal lines of the calibration image (brown) and average deviations of points from 200 randomly-generated straight image lines.**



**Figure 32: Histograms of distributions of average deviations of points from the projected horizontal lines (yellow) and vertical lines (brown) of the calibration image and average deviations of points from 200 randomly-generated straight image lines (blue).**

#### **4.3.4 The Geometric Distortion-correction Process**

The projected horizontal lines of the calibration image have been shown to be the most distorted group. They will therefore be candidates for rectification. Two different methods may arise for carrying out the selection of lines for rectification. The first would generally rectify all horizontal lines and ignore the vertical ones, and the second will set a threshold distortion value for all lines and those lines whose distortion values go above the threshold will be rectified while other lines will not be rectified. The first method of line rectification used in this work rectifies only the horizontal lines of the calibration image. Repeated distortion-correction tests on the static projection surface using this method show that vertical lines of the calibration image remain relatively undistorted before and after the correction. The results of the various geometric distortion-correction tests carried out with the calibration image are presented in Section 4.4.

The process of distortion correction of static non-planar projection surfaces using a calibration image is outlined as follows:

(i) Generate the calibration image with an appropriate number of feature points with known horizontal and vertical pixel coordinates and project it onto the surface. Normalise the pixel coordinates of its feature points.

(ii) Capture the image of the projection with the camera.

(iii) Use a suitable corner point detector to obtain the required feature points of the captured image and arrange them in rows and columns according to the order in which they appear in the original calibration image. Also normalise the pixel coordinates of these feature points.

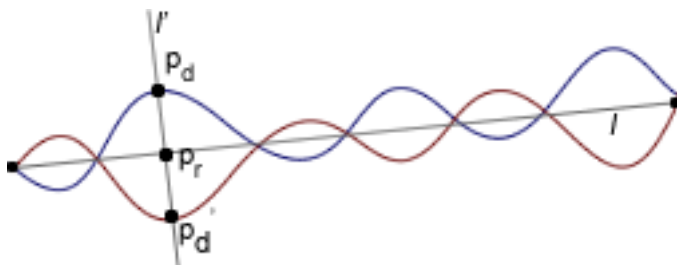
(iv) With the coordinates of the feature points arranged in their correct positions, form a series of horizontal and vertical lines by linking together the first and last points of each row and column formed by feature points. Also form the desired undistorted quadrilateral by linking the four extreme feature points of the captured image.

(v) Calculate the parameters of each horizontal line and the deviation of each of its member points from the line as described in Section 3.4.2. Also calculate the parameters of each vertical line and the deviation of each of its member points from their respective lines. Also calculate the average deviation value for each line and set a threshold for distortion if desired. If a threshold is set, make a decision to correct the geometric distortion of those lines whose average deviation values are equal to or above the set threshold. If no threshold is set, correct the distortion of only the horizontal lines.

(vi) Use the parameters of each line calculated in (v) to calculate the desired observed coordinates of each member feature point on the captured image of the projection. The desired coordinates of a distortion-corrected feature point on the distortion-corrected line will be the coordinates of the point of intersection between it and a perpendicular line drawn from the distorted feature point. The distortion-corrected feature point also coincides with the midpoint of the perpendicular line drawn from the distorted feature point to another point opposite the distorted feature point and equidistant to the distortion-corrected line.

Let the distorted feature point be labelled as  $P_d$ , the distortion-corrected point as  $P_r$ , and the point equidistant to the distortion-corrected line  $l$  but opposite  $P_d$  be  $P_d'$ .

Let the line perpendicular to  $l$  be  $l'$ . The illustration given in figure 33 shows the positions of the points and lines so described.



**Figure 33: Illustration of the respective positions of the distortion-corrected line  $l$ , a line  $l'$  perpendicular to it, and distorted feature point  $p_d$ , opposite point  $p_d'$  and distortion-corrected point  $p_r$ . Point  $p_r$  is the midpoint between  $p_d$  and  $p_d'$  and also the point of intersection of lines  $l$  and  $l'$ .**

(vii) Use the four extreme feature points of the captured image of the projection and the four corresponding feature points of the calibration image to estimate a desired 2D homography between both images. This desired homography defines the 2D mapping of distortion-free feature points from the captured image of the projection to the feature points of the calibration image.

The homography may also be estimated by using more than four feature points from both images but with an optimisation method used to determine the best overall transformation.

(viii) The next task will be to devise a way to warp the calibration image in order to observe a geometrically undistorted projection of the image. The warping scheme used is derived from that developed in Section 4.2 where the directions of warping of the lines of the calibration image are opposite to the respective directions of the observed distortions of these lines. The 2D homography estimated in step (vii) is used to transform member feature points  $\mathbf{p}_d'$  of each distorted line to respective control points  $\mathbf{w}_d'$  on the calibration image. The distortion-corrected points  $\mathbf{p}_r$  are transformed by this homography to points  $\mathbf{w}_r$  on the calibration image. These points should ideally coincide with the known feature points of the calibration image, but do not always do so as a result of projection errors.

To complete the warping of the calibration image, points  $\mathbf{w}_r$  are warped to control points  $\mathbf{w}_d'$ .

(ix) Estimate the RBF coefficients using the method outlined in Section 3.6 where the set of coordinates of  $\mathbf{w}_r$  are the initial control point positions and the set of coordinates of  $\mathbf{w}_d'$  are final control point positions.

(x) Use the estimated RBF coefficients to warp the whole calibration image as described in Section 3.6.

(xi) Finally substitute the calibration image with a natural image and warp the entire natural image using the same estimated RBF coefficients. Care must be taken to ensure that the camera, projector, and projection surface remain fixed in position and the projection surface does not deform within the time leading to the estimation of the RBF coefficients

and the time of substitution with the natural image, otherwise the process of distortion measurement with the calibration image and substitution with the natural image will have to be repeated for the new positions of the projector, camera and projection surface.

The geometric distortion-correction procedure described above was used to correct the static non-planar projection system described for various poses of the projector and camera and various shapes of the projection surface. Image-matching tests including NCC and Average Deviation Errors were used to objectively measure the effectiveness of the distortion-correction process. The following parameters were varied during the tests conducted on the static non-planar projection system:

(i) The number of feature points on the calibration image: This was done to determine the optimum number of control points needed to carry out effective geometric distortion correction with the calibration image. Distortion correction was carried out for calibration images with a number of feature points ranging from 36 (6 horizontal x 6 vertical) to 196 (14 horizontal x 14 vertical) while keeping the positions of the projector and camera, and the position and shape of the projection surface fixed.

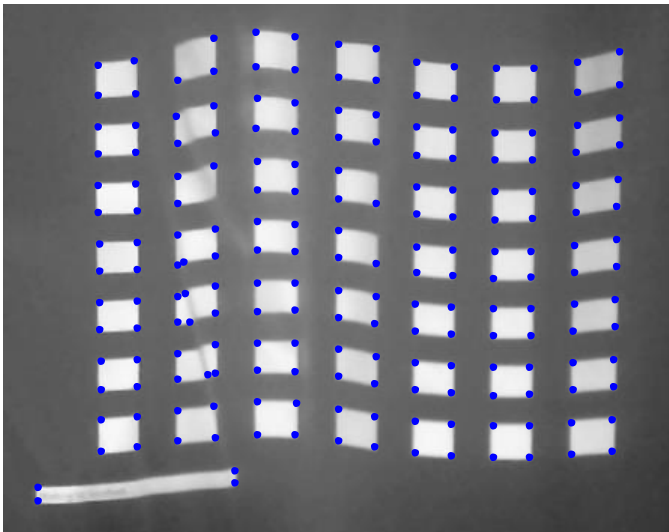
(ii) The type of RBF used for warping: Three RBFs were compared in these tests namely, Multiquadric, Inverse Multiquadric and the Gaussian to determine the RBF with the best performance in terms of least deviation errors and average time taken to warp an image of size 640 x 480 pixels.

(iii) The pose of the projector and camera: The positions of the projector and camera and the shape of the projection surface were changed from one test to the next in order to investigate the effectiveness of the geometric distortion-correction methods for various projector/camera poses and different observed levels of distortion. The average deviation of points from each horizontal and vertical line was calculated for the distorted and distortion-corrected calibration image projections. The calibration image was substituted with a natural image and the NCC for the distorted and corrected images were obtained and compared. The whole process was repeated after changing the positions of the

projector, camera and the projection surface. The results of these tests are presented in Section 4.4.

#### 4.3.5 Experimental Considerations

(i) Detection and arrangement of image feature points: The success of this geometric distortion-correction system is largely dependent on the correct detection and arrangement of corners of the calibration image. Sometimes during the projection of the calibration image some true corner points may not be detected by the corner point detector while other false corner points may be detected by the detector. Both problems are illustrated in figure 34.



**Figure 34: Image of a calibration image showing correctly detected feature (corner) points, missing and unwanted feature points**

It was observed in this work that sufficient illumination of the projection environment, elimination of creases and unwanted shadows on the projection surface reduced the chances of missing and unwanted corner points. Where such problems are inevitable, the algorithm designed to arrange these points in the correct order must be robust enough to recognise unwanted and missing feature points.

(ii) Effects of Noise: The effect of noise on the distortion-correction system was not investigated. It is evident that added noise hampers the accurate detection and matching of feature points and subsequently negatively affects the distortion-correction results.

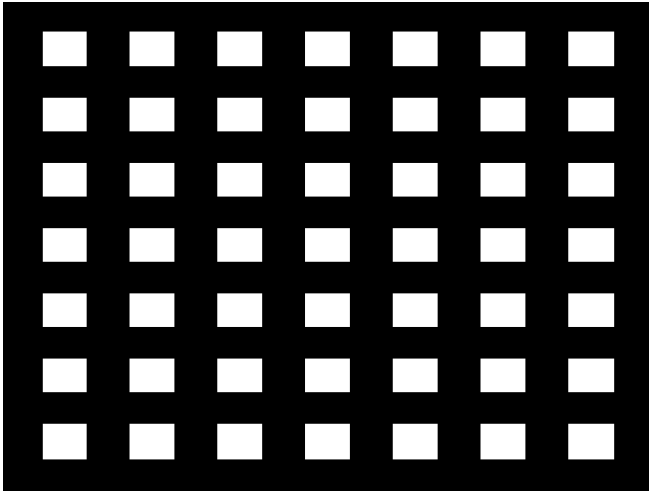
However the visual quality of the images of the projection acquired in all tests served the distortion-correction purpose.

(iii) Visibility of the whole projection surface: Another precaution observed while testing the distortion-correction system was making sure the whole projection surface was visible to the camera and without objects or markings on the projection surface capable of influencing the detection of unwanted feature points.

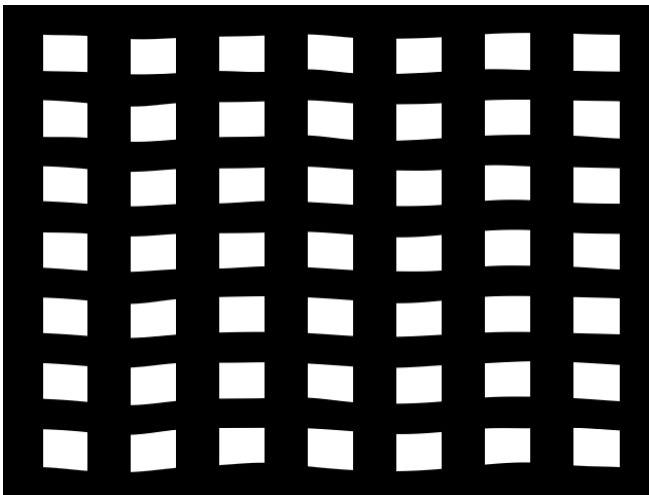
#### **4.4 Results of Distortion-correction Tests with the Calibration Image**

##### **4.4.1 General Distortion-correction Results**

The first set of results presented in figures 35 and 36 were obtained using a calibration image consisting of 196 feature points forming 14 horizontal and 14 vertical lines and warping done with the multiquadric RBF. The figures are arranged in groups of four. Each group consists of the non-distorted and pre-warped pair of projected images and the respective camera images of their observed projections. They show that the geometric distortion of the observed projection of the calibration image and the natural images was achieved after pre-warping both images before projecting them onto the non-planar surface.

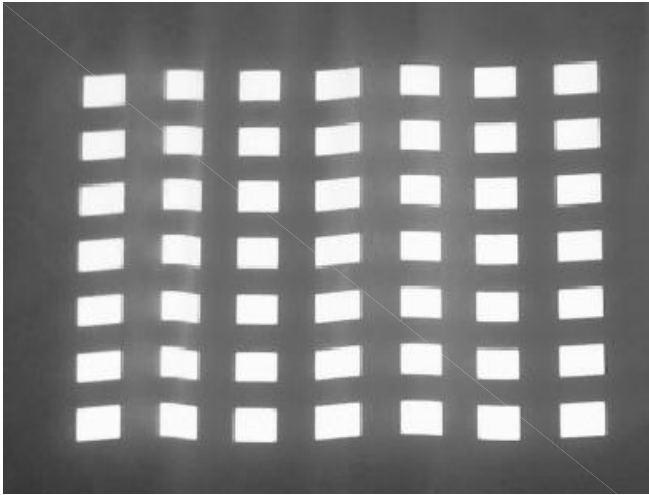


**(a) Non-distorted calibration image with 196 feature points projected onto the projection surface**

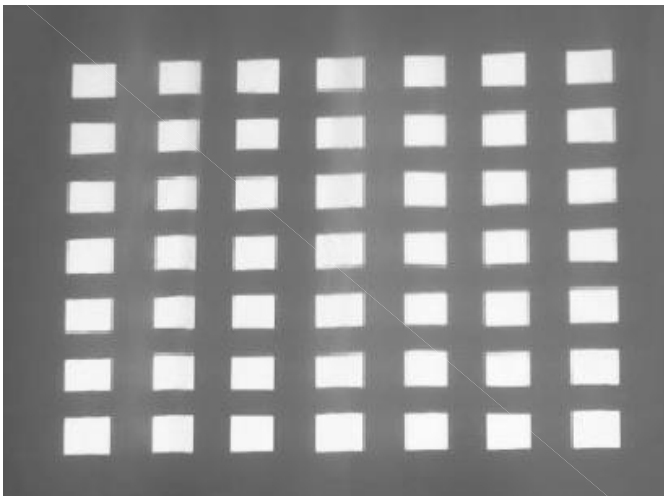


**(b) Pre-warped calibration image projected onto the non-planar projection surface**



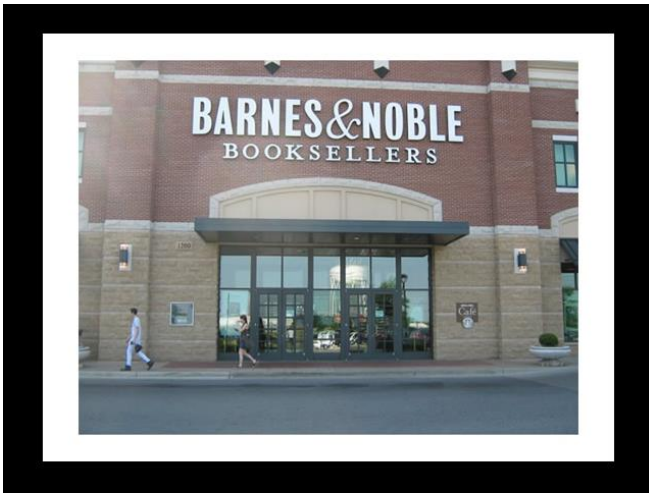


**(c) Observed distorted projection of the non-distorted calibration image**

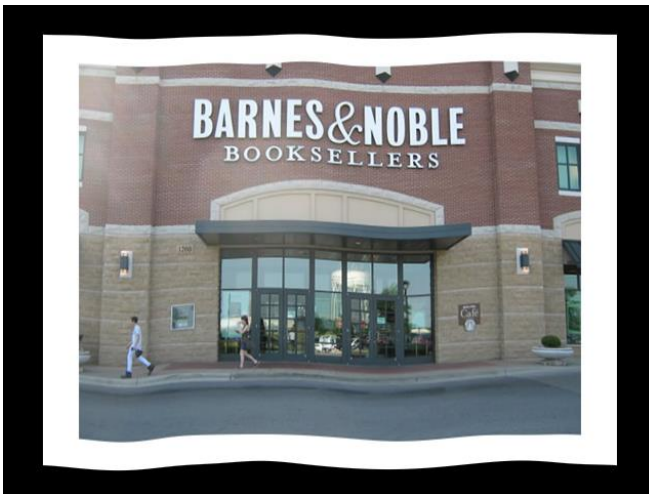


**(d) Observed distortion-corrected projection of the pre-warped calibration image**

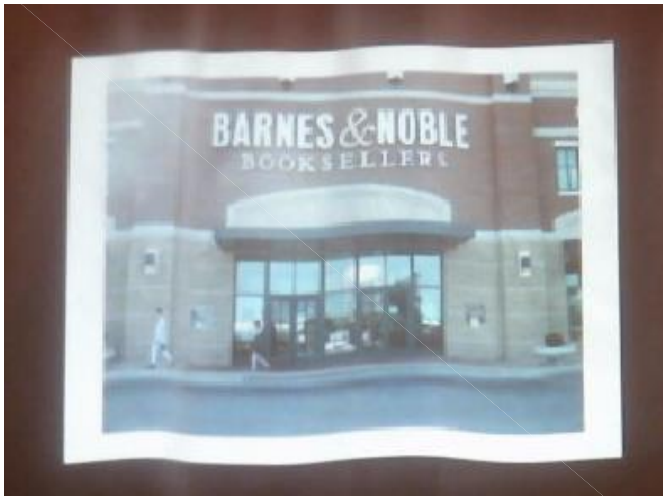
**Figure 35: Geometric distortion correction of the projection of a calibration image consisting of 196 feature points. (a) and (b) show the non-distorted and pre-warped calibration image while (c) and (d) show the corresponding observed distorted and distortion-corrected projections of the calibration image.**



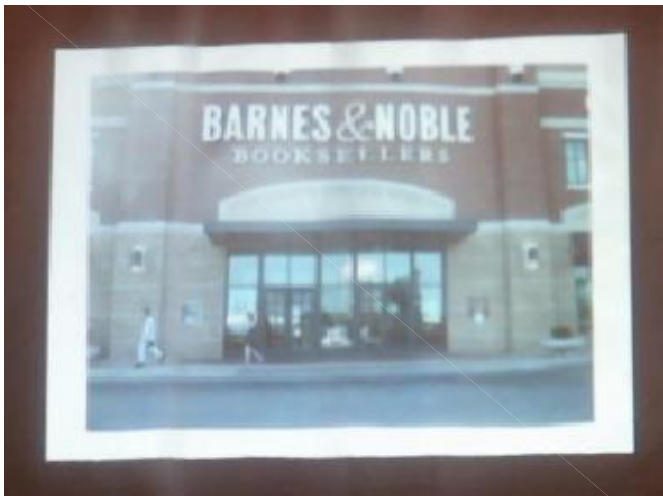
**(a) Non-distorted projected image**



**(b) Pre-warped projected image**



**(c) Observed distorted projection of the natural image**

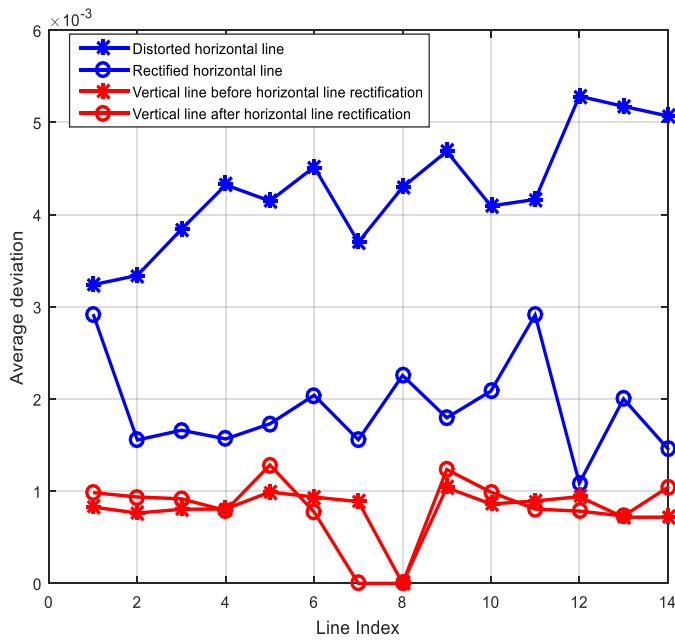


**(d) Observed distortion-corrected projection of the projected natural image**

**Figure 36: Geometric distortion correction of the projection of a natural image replacing the calibration image and using the warping coefficients obtained from the calibration image. (a) and (b) show the non-distorted and pre-warped image while (c) and (d) show the corresponding observed distorted and distortion-corrected projections of the natural image.**

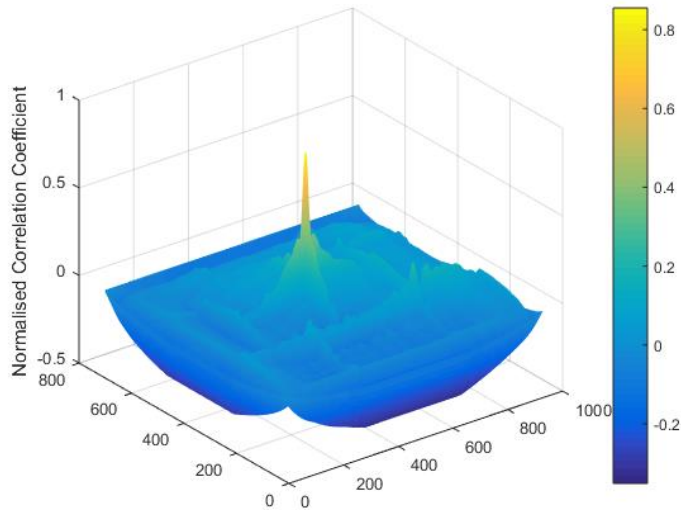
The results presented in figure 37 show the average deviation errors of feature points from each horizontal and vertical line of the observed projection of the calibration image before and after the distortion-correction process was carried out. Average deviation values were reduced by as much as 78.8% for horizontal lines. This represents a significant improvement for the horizontal lines. The deviation values for the vertical lines show that the distortion-

correction process improved the straightness of some lines with lower average deviation values but worsened the straightness of others. The average deviation values of the affected vertical lines however still remained significantly lower than those of the least distorted horizontal lines, proving that the distortion-correction process for horizontal lines did not significantly distort the vertical lines.

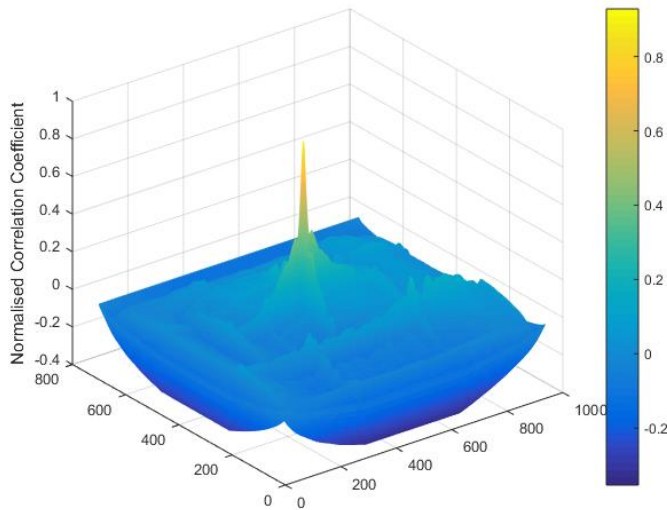


**Figure 37: Average deviation of feature points from their respective horizontal and vertical lines on the observed images of projection of the calibration image (figures 35(c) and 35(d)) before and after geometric distortion correction. Average deviation values of points for distortion-corrected lines are lower than the average deviation values for distorted lines.**

The results for the Normalised Correlation Coefficient (NCC) test for both distorted and distortion-corrected images of the projection in figures 36(c) and 36(d) are shown in figures 38(a) and 38(b) respectively. An increase in the peak NCC value from 0.855 to 0.928 shows that the geometric distortion-correction process improved the match between the distortion-corrected image of the projection and its desired non-distorted image.



**(a) NCC for distorted observed image of the projection. Peak NCC value is 0.855.**

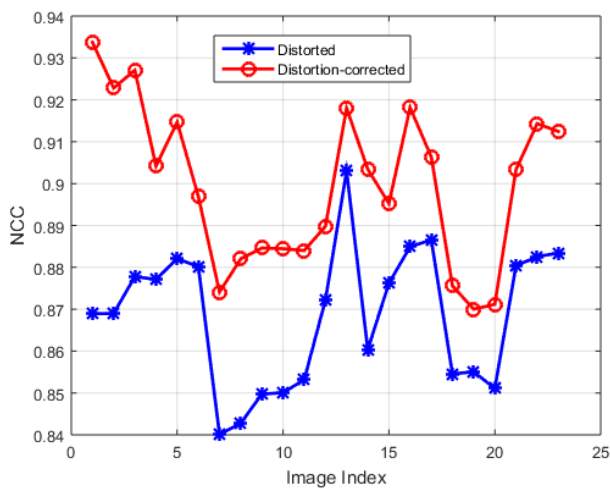


**(b) NCC for distortion-corrected observed image of the projection. Peak NCC value is 0.928.**

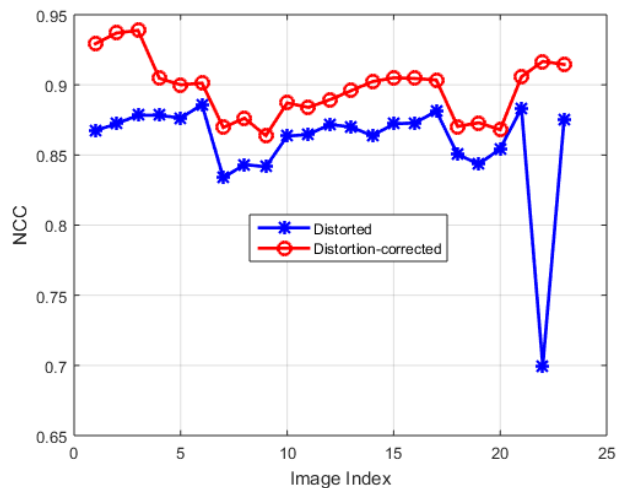
**Figure 38: Normalised Correlation Coefficients (NCC) of the observed distorted and distortion-corrected projections. The NCC was calculated using the cropped non-distorted homography-transformed image of figure 36(a) as template and the distorted and distortion-corrected observed images of its projection of figure 36(c) and 36(d) respectively. An improvement in NCC indicates a better match between the template and the distortion-corrected image than the template and the distorted image, also revealing satisfactory distortion-correction.**

Figure 39 (a)-(c) shows sets of peak NCC values obtained from distortion correction tests performed using calibration images with 100, 144 and 196 feature points with the multiquadric, inverse multiquadric and Gaussian RBFs used for warping for various poses of the projector and camera and shapes of the projection surface.

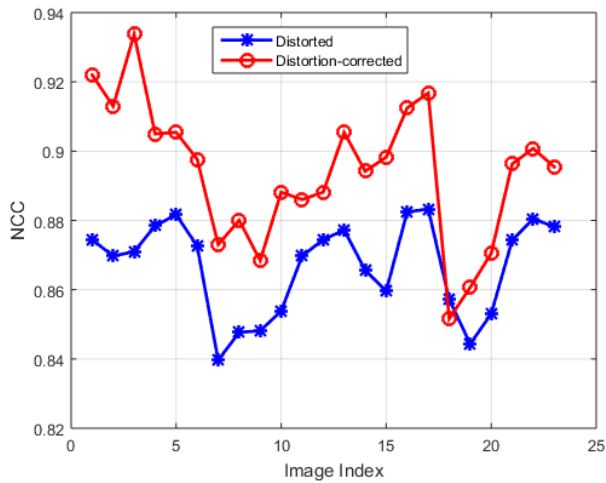
The red lines in the plots show an improvement in peak NCC for distortion-corrected images over distorted images represented by the blue lines, by as much as 1.66% to 7.45% with 196 feature points, 1.58% to 31.03% with 144 feature points, and -0.66% to 7.20% with 100 feature points. The negative sign indicates a lower peak NCC for the distortion-corrected image than that of the distorted image. The results also demonstrate that geometric distortion correction can be achieved with calibration images having an appropriate number of feature points and warping with any of the three RBFs considered. The results of tests performed to determine the appropriate number of feature points for the calibration image and the type of RBF to use for warping are presented in the following subsections.



**(a) 196 feature and control points**



**(b) 144 feature and control points**

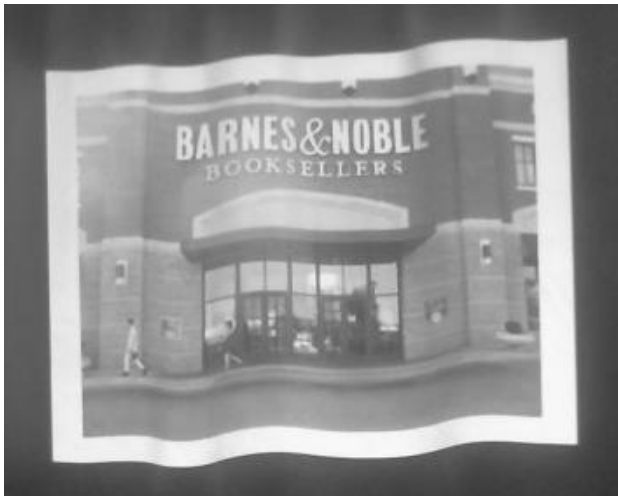


**(c) 100 feature and control points**

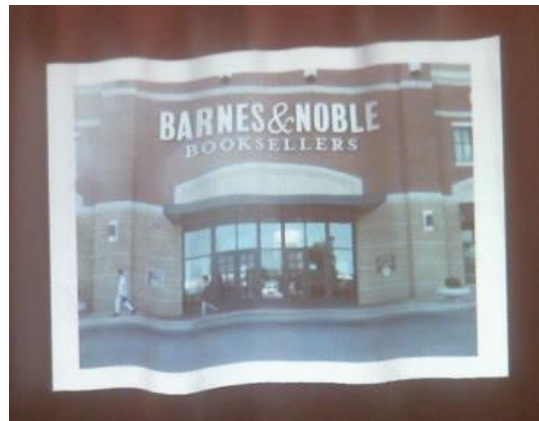
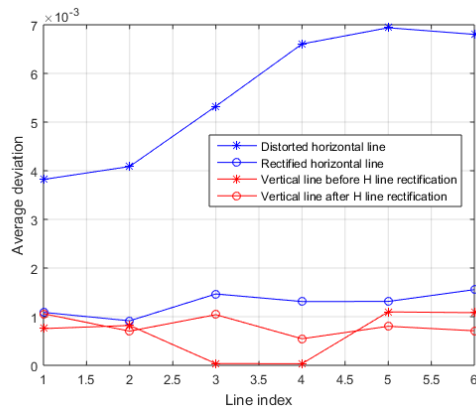
**Figure 39: Peak NCC values for distorted and distortion-corrected projections using 196 (a), 144 (b), and 100 (c) feature points for the calibration image and 3 different RBFs for warping**

#### **4.4.2 Results of Tests to Determine the Number of Feature Points to Use in the Calibration Image**

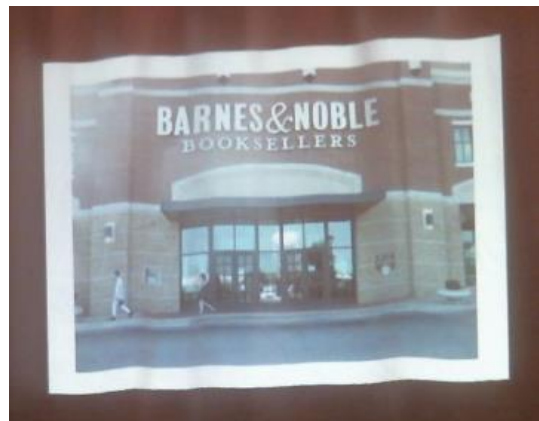
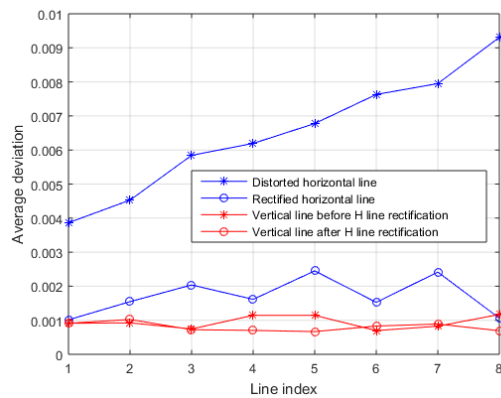
By varying the number of feature points of the calibration image from 36 (6 rows x 6 columns) to 196 (14 rows x 14 columns) in performing geometric distortion correction, it was observed that while the calibration image lines were being straightened with fewer number of feature points, the substituted natural image still suffered significant geometric distortion as shown in the images of figure 40. Using more feature points for the calibration image is shown here to produce better distortion-correction results because more points produce a better sampling and representation of the distortion of the surface. Using more points would however impact negatively on the speed of the warping process as more computations would be involved.



(a) Distorted output image

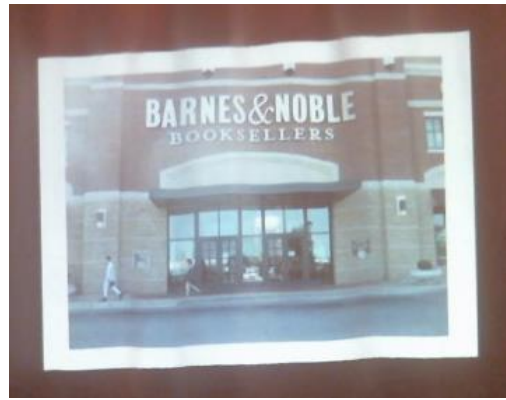
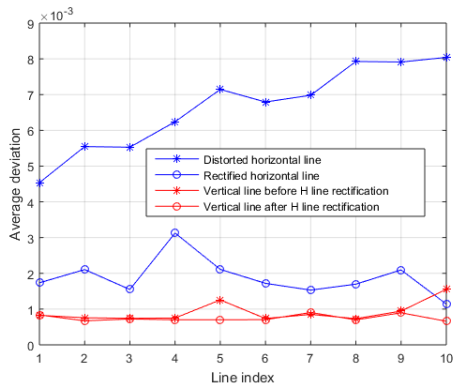


(b) Average deviation plots for 36 feature points and corresponding insufficiently distortion-corrected image

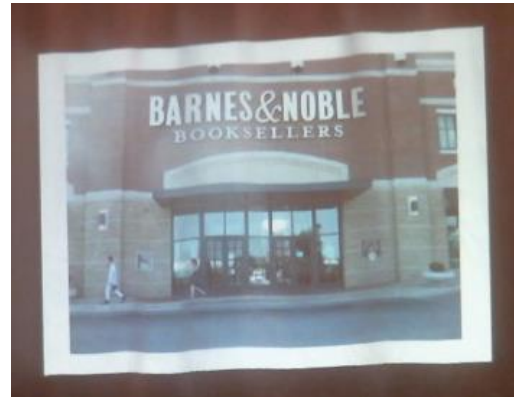
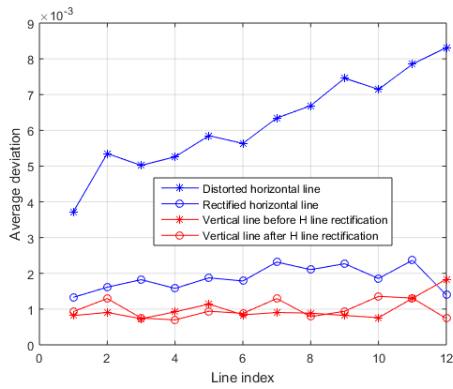


(c) Average deviation plots for 64 feature points and corresponding insufficiently but slightly better distortion-corrected image

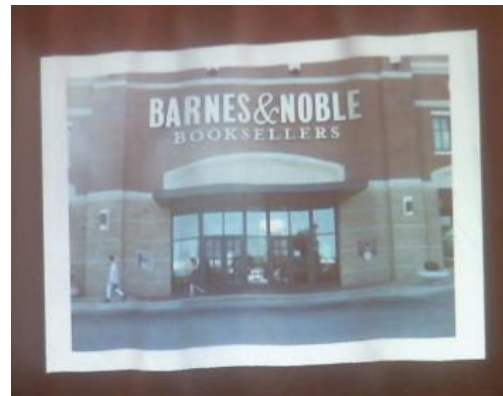
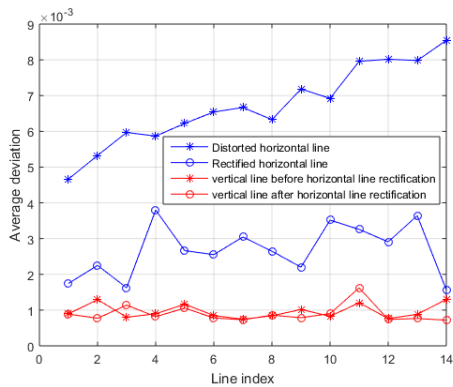




**(d) Average deviation plots for 100 feature points and corresponding better distortion-corrected image**



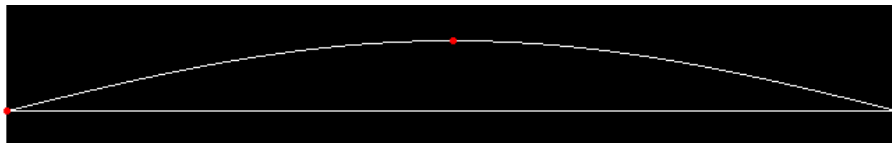
**(e) Average deviation plots for 144 feature points and corresponding better distortion-corrected image**



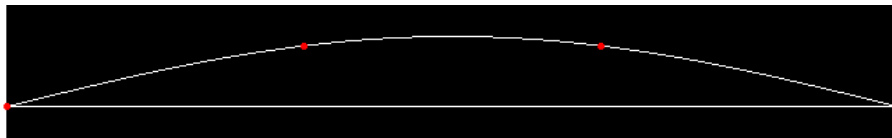
**(f) Average deviation plots for 196 feature points and corresponding better distortion-corrected image**

**Figure 40: Results to demonstrate the need for having a sufficiently high number of feature points for the calibration image to achieve better geometric distortion correction.**

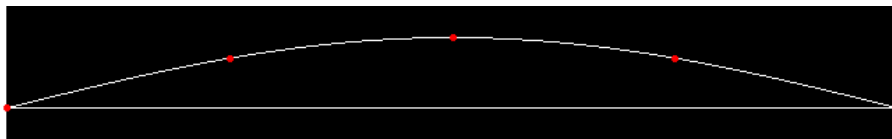
A test to determine an optimum number of feature points per line of the calibration image was carried out with a typical section of a composite curve containing three points of inflexion: two points at each end (where the section would link to another) and one crest (turning point). The section so described is shown in figure 41. It involved assigning a number of feature points to be used as RBF control points to warp the curve into a straight line and calculating the average deviation of the resulting line from the desired line. The feature points dividing the section of a composite curve into 3, 4, and 5 parts respectively are shown as red dots in figures 41(a), (b) and (c). Average deviation values for each of them were calculated to be  $1.405 \times 10^{-16}$ ,  $1.1238 \times 10^{-16}$ , and  $9.1154 \times 10^{-17}$  respectively revealing that the deviation value decreases with the number of feature points used. More importantly, the negligible deviation values show that each section of the composite curve should have at least 3 evenly-distributed feature points.



**(a) Section of composite curve with 3 feature points**



**(b) Section of composite curve with 4 feature points**



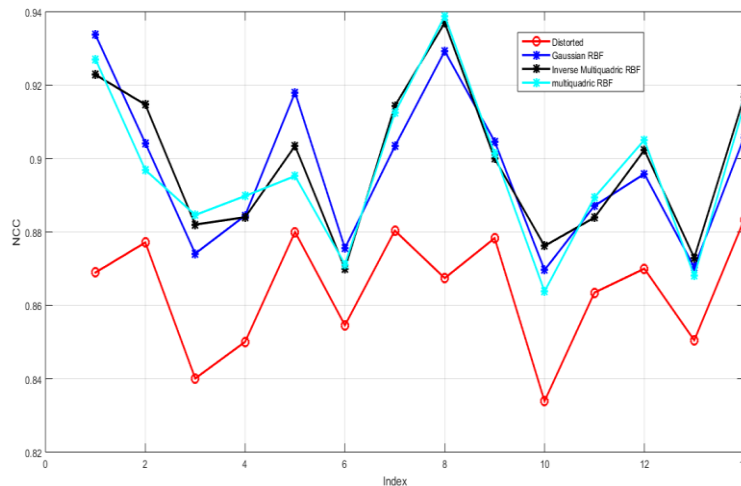
**(c) Section of composite curve with 5 feature points**

**Figure 41: Curve sections with varying number of feature points to be used as RBF control points to warp the curve sections into the straight horizontal lines**

The shape of the projection surface determines the number of feature points needed for the calibration image for effective distortion correction. A calibration image with 196 feature points was found to be sufficient in this work for all shapes of the projection surface while 100 (10 rows x 10 columns) and 144 (12 rows x 12 columns) feature points were sufficient in most cases. Using more than 196 feature points caused instability in the distortion-correction process by introducing falsely-detected feature points.

#### 4.4.3 Results of Tests to Determine the Type of RBF to Use for Distortion Correction

The Gaussian, multiquadric and inverse multiquadric RBFs were compared based on the quality of geometric distortion correction they produced (represented as peak NCC values) for calibration images consisting of 144 and 196 feature points and various poses of the projector and camera. The results in figure 42 shows that no one RBF performed outstandingly better than the other two, as they each exhibited relatively high and low peak NCC values in different tests.



**Figure 42: Comparison of three RBF types based on the NCC values of distortion-corrected images**

The 3 RBF types were also compared on Matlab running on Windows 10 OS with an Intel Celeron N2830 2.16GHz processor based on the average computation time from 5 trials required to warp a  $640 \times 480$  pixel image with 196 control points. The result in Table 4

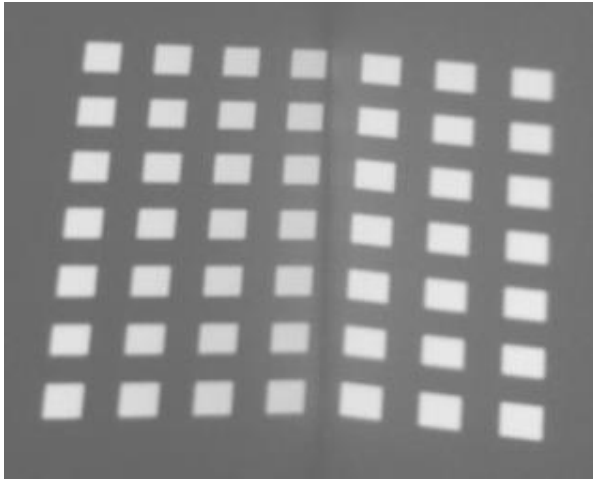
shows that the multiquadric RBF performed best with the lowest average computation time of 19.9716s. This is attributed to the relatively short time taken by the processor to perform a square-root operation compared with the inverse-of-a- square-root and exponential operations.

**Table 4: Comparison of three RBF types based on the average computation time to warp a  $640 \times 480$  pixel image on Matlab**

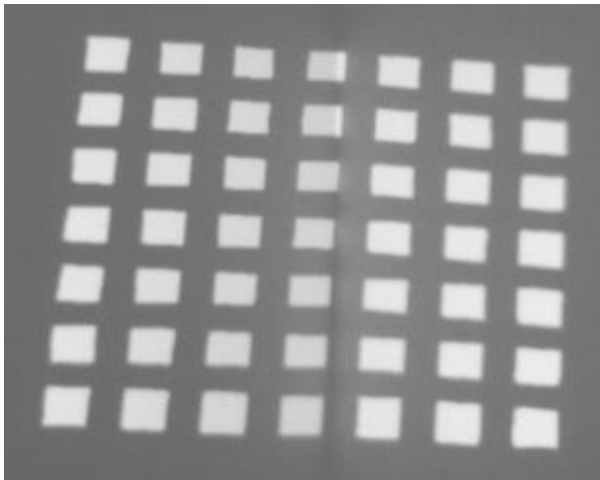
<b>RBF type</b>	<b>Average Computation time using Matlab (s)</b>
Gaussian	23.3876
Inverse Multiquadric	22.7238
Multiquadric	19.9716

#### **4.4.4 Results of Distortion-correction Tests for Other Non-planar Surfaces, eg Two Planar Surfaces Meeting at Right Angles**

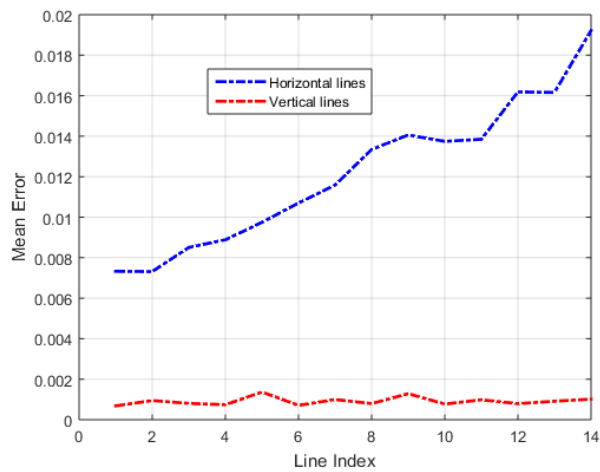
The geometric distortion correction process was also tested on a static non-planar projection surface consisting of two planar surfaces (walls of a room) meeting at right angles. The results presented in figure 43 also shows that geometric distortion correction using this method performed well with improved average deviation values and higher peak NCC values for images of the distortion-corrected projection than for images of the distorted projection.



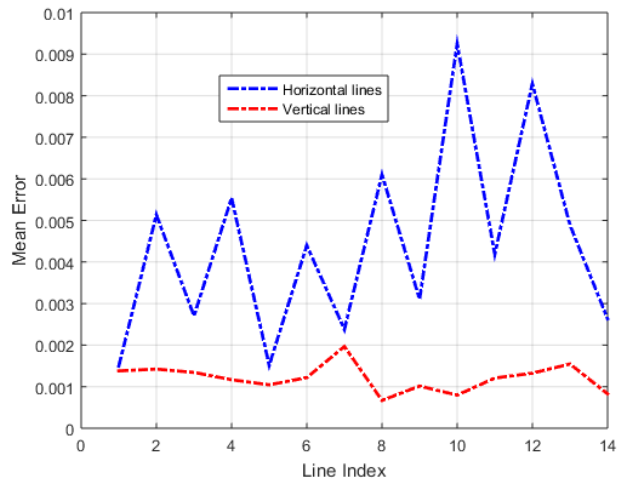
**(a) Observed distorted projection of the non-distorted calibration image**



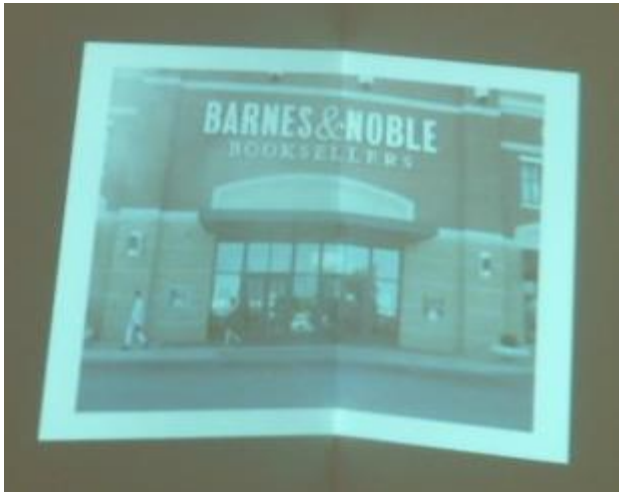
**(b) Observed distortion-corrected projection of the pre-warped calibration image**



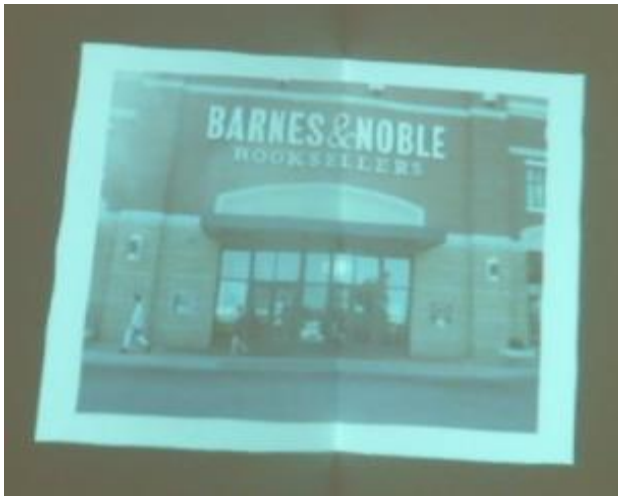
**(c) Average deviation plots of feature points from their respective horizontal and vertical lines on the observed images of projection of the non-distorted calibration image**



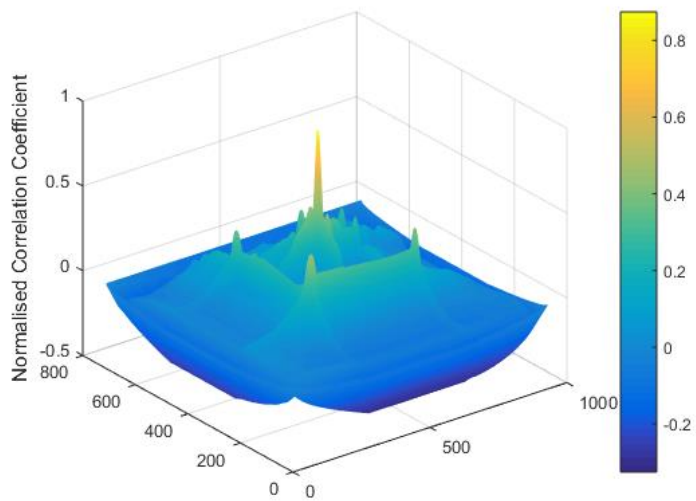
**(d) Improved average deviation plots of feature points from their respective horizontal and vertical lines on the observed images of projection of the pre-warped calibration image**



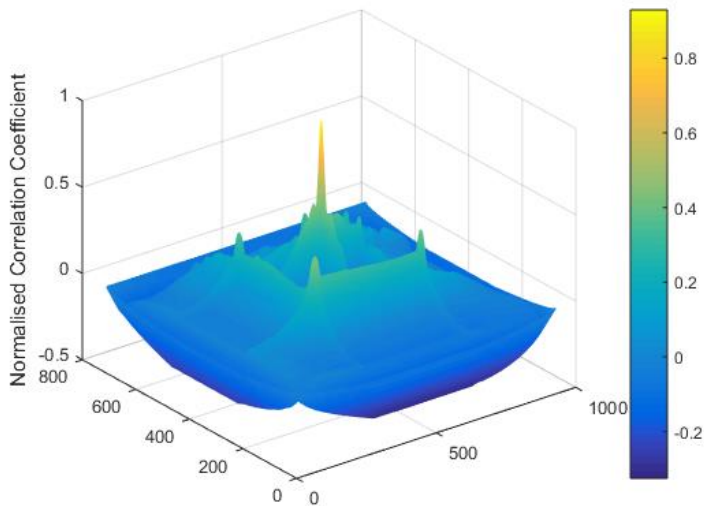
**(e) Observed distorted projection of the natural image**



**(f) Observed distortion-corrected projection of the natural image**



**(g) NCC values for distorted observed image of the projection. Peak NCC value is 0.874.**



**(h) NCC values for observed distortion-corrected image of the projection. Peak NCC value is 0.929.**

**Figure 43: Distortion-correction results for two planar surfaces meeting at right angles**

#### **4.5 Automatic Geometric Distortion-correction Without Using a Calibration Image**

The limitation in using a calibration image to estimate geometric distortion before correcting it is that if the surface changes shape or deforms, or the camera or projector change position, the calibration image will have to be projected again in order to update the system to produce desirable results. This type of distortion correction will be very difficult to implement in video projection systems because the streaming video will have to be interrupted for recalibration of the system with the calibration image to take place. It will also be a practically impossible distortion correction technique for projection systems that use deforming surfaces as it will be impossible to achieve a seamless projected display while tracking the shape of the projection surface in real time with a calibration image. A scheme for distortion correction that would eliminate the need for projecting the calibration image is therefore important in these cases. Three methods of correcting distortion without the use of a calibration image were tested in this work. The first two methods rely on detecting and matching corresponding features of the projected natural image and the image of its projection captured by the camera to be used in estimating the RBF coefficients to pre-warp the projected image.



The third method relies on the characteristics of the projection surface. This method states that if the projection image is to be warped along defined grid lines to correct the distortion of the observed display, then the deviation of the coordinates of the points along these lines is assumed to be a linear function of their respective positions on these lines. The description of each technique tested is presented in the following subsections. In all the techniques  $w_r$  and  $w_d'$  respectively represent the initial and final warped coordinates of the control points of the projected natural image.

#### **4.5.1 Edges and Corners at the Periphery of the Projected and Captured Images Used as Feature Points for Distortion Correction**

For this method, the edges at the periphery of the camera image of the projection were detected and sampled. The four corner points at the extreme corners were also detected. The straightness of the pair of horizontal edges at the top and bottom of the image was used as criterion for correction of distortion. In this test, 14 points at the top and another 14 at the bottom distorted edges were used. The correction scheme used is similar to that described in Section 4.3.4, the only differences being that the feature points used are not transformed points from a calibration image and the RBF coefficients so obtained are from a pair of distorted lines. The particular steps used in this method are outlined as follows:

(i) The top and bottom edges of the projected image are divided into the required number of points with normalised pixel coordinates. 14 points were used in this work.

(ii) An appropriate edge detector and image segmentation technique is used to extract the top and bottom edges of the captured image of the projection on the projection surface. A suitable corner detector is also used to obtain the coordinates of the four extreme corner points of the captured image. All feature point coordinates are normalised.

(iii) The four corner points so obtained are linked to form the desired undistorted quadrilateral. The top and bottom edges of this quadrilateral are the straight lines to which the correction of the distorted projection is based.

(iv) The homography between the projected image and the undistorted quadrilateral is then estimated using the four pairs of matching corner points and all points obtained in step (i) are transformed with this homography to respective points lying on the top and bottom edges of the desired undistorted quadrilateral.

(v) The parameters of the top and bottom horizontal lines of the undistorted quadrilateral are calculated and these parameters are used to calculate the coordinates of distorted points that coincide with the distorted edges from the coordinates of the undistorted ones. This step is similar to step (vi) in Section 4.3.4 and illustrated in figure 33. In this case however, the desired feature point  $\mathbf{p}_d$  has to be determined from a group of points that form the undistorted edge. Points  $\mathbf{p}_d'$  and  $\mathbf{p}_r$  are also obtained in the same way as described in Section 4.3.4 and illustrated in figure 33.

(vi) The inverse homography estimated in step (iv) is used to transform the coordinates of distorted points  $\mathbf{p}_d'$  to  $\mathbf{w}_d'$  and the RBF warping coefficients required for transforming  $\mathbf{w}_r$  to  $\mathbf{w}_d'$  are also calculated.

(vii) The RBF coefficients thus obtained are used to warp the entire projected image. The results of this warping process are presented in Section 4.6.1.

#### **4.5.2 Blobs in the Projected and Captured Images Used as Feature Points for Distortion Correction**

The method of using blob features matched from the projected image and the distorted image of the projection attempts to restore image points to their original non-distorted positions. RBF warping coefficients can be calculated using the coordinates of these points. The Speeded-Up Robust Features (SURF) implementation of SIFT was used to detect blobs in both images and match them with each other. Feature points along the periphery of the distorted image obtained in Section 4.5.1 were also added to the feature points obtained from SIFT.

Precautions taken in matching these points to minimise the harmful effects of mismatching feature points include the following:

(i) Using a maximum distortion threshold after transforming the camera image to the projected image with the homography. The distortion threshold used was the maximum absolute distance calculated between the set of  $w_r$  and  $w_d$  points.

(ii) Points too close to each other were excluded to prevent singularity in the estimation of the RBF transformation matrix  $A$  of equation 39. Where there is a cluster of points only a pair of matching points is chosen.

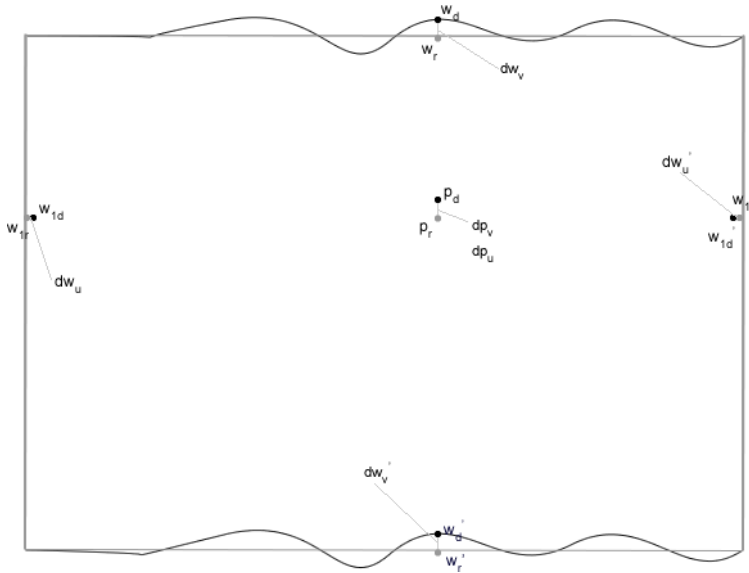
(iii) All matched points falling outside the boundary of the projected image were excluded as they do not represent any significant features. The results of this correction are presented in Section 4.6.2.

### **4.5.3 Linear Modelling of Distortion**

The results presented in Sections 4.6.1 and 4.6.2 relating to automatic geometric distortion correction by using only edges and corners of the periphery of the image and/or using blob features (derived from the SIFT algorithm) as feature points show that both methods are inadequate in solving the problem because of the insufficient number and distribution of the feature points they present around the distorted image.

The method of linear modelling of geometric distortion assumes that the series of 2D curves on the surface that cause the distortion of straight lines projected onto the surface observed as a series of crests and troughs, is uniform from top- to- bottom and left-to- right ends of the surface. A close observation of previous results of the trend of warping of the projected image onto the surface to correct the observed distortion show that the warping trend appears uniform horizontally and vertically. Using this observed uniformity, a linear model relating the horizontal and vertical coordinates of respective control points to the amount of displacement they go through in the warped projected image can be applied to obtain a sufficient number of well-distributed control points all over the image.

The illustration in figure 44 will be used to explain this concept.



**Figure 44: Illustration of a warped projected image showing the initial edges (black), the final edges after warping and the initial and final points along each edge.**

Figure 44 represents a natural image to be pre-warped, projected, and observed undistorted on the projection surface. The coordinates of points are calculated as explained in previous Section 4.5.1. According to figure 44  $w_r$  and  $w_r'$  are corresponding points on the top and bottom edges of the unwarped projected image respectively, and  $w_d$  and  $w_d'$  are respective points to which they would be transformed after warping the image.

$w_{1r}$  and  $w_{1r}'$  are corresponding points on the same horizontal line and lying on the left and right edges of the unwarped projected image respectively.

$w_{1d}$  and  $w_{1d}'$  are respective points to which they are transformed after warping the image. The difference between the vertical coordinates of  $w_d$  and  $w_r$  and  $w_d'$  and  $w_r'$  is given as  $dw_v$  and  $dw_v'$  respectively while the difference between the horizontal coordinates of  $w_{1d}$  and  $w_{1r}$  and  $w_{1d}'$  and  $w_{1r}'$  is  $dw_u$  and  $dw_u'$  respectively.

The task is to determine the horizontal and vertical coordinates of point  $p_d$  resulting from warping of the already-known coordinates of the point  $p_r$  that lies between extreme points  $w_r$  and  $w_r'$  and  $w_{1r}$  and  $w_{1r}'$ . To do this, a linear relationship is assumed between the

coordinates of the extreme points and the displacement they suffer after warping in the form

$$dp_u = a_u p_u + b_u \text{ and } dp_v = a_v p_v + b_v \quad (58)$$

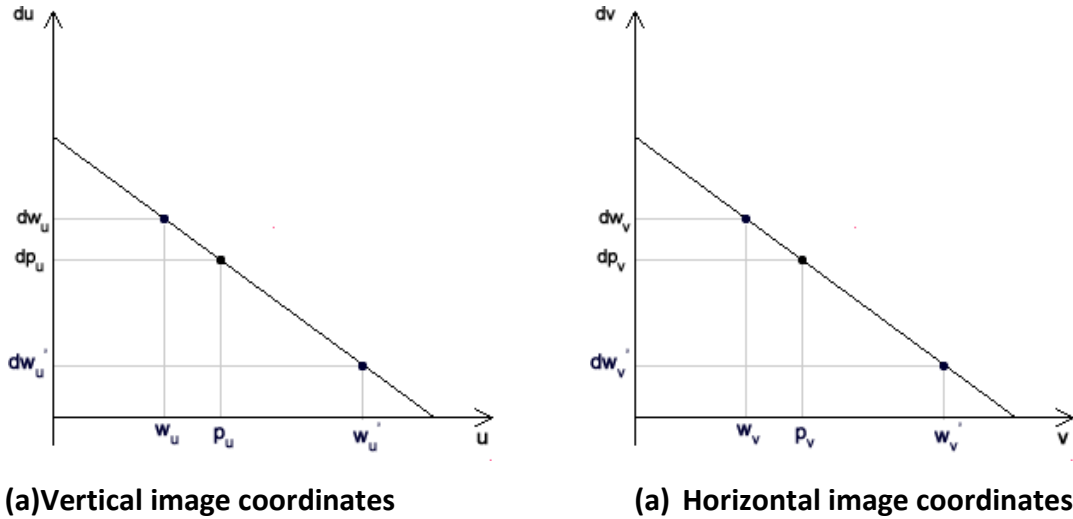
$$\text{where } a_u = \frac{(dw'_u - dw_u)}{(w'_u - w_u)}, b_u = dw_u - w_u \frac{(dw'_u - dw_u)}{(w'_u - w_u)}$$

$$\text{and } a_v = \frac{(dw'_v - dw_v)}{(w'_v - w_v)}, b_v = dw_v - w_v \frac{(dw'_v - dw_v)}{(w'_v - w_v)}$$

The final coordinates of the warped point

$$p_u' = p_u + dp_u, p_v' = p_v + dp_v \quad (59)$$

The assumed linear relationship between the coordinates of the points in the unwarped image and their respective displacements in the warped image is illustrated in figure 45. The results of the distortion-correction tests using this method to obtain control points inside the image for warping are given in Section 4.6.3.



**Figure 45: Illustration of the assumed linear relationships between coordinates of points in the unwarped and warped images.**

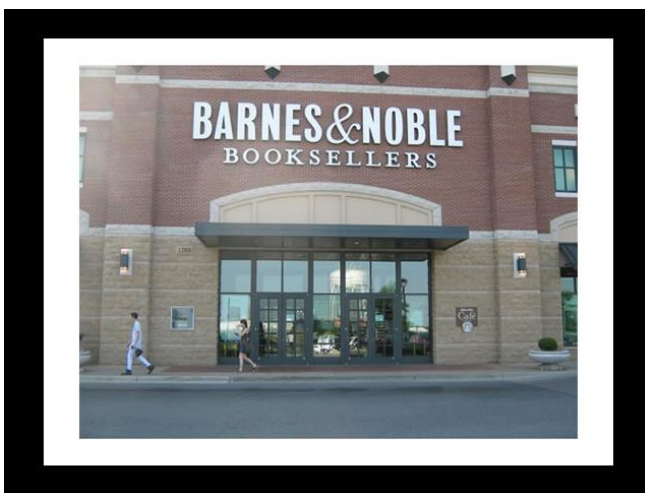
It is observed that very negligible changes occur in the horizontal coordinates from the unwarped to the warped images.

## **4.6 Results of Tests of Automatic Geometric Distortion-correction Without Using a Calibration Image**

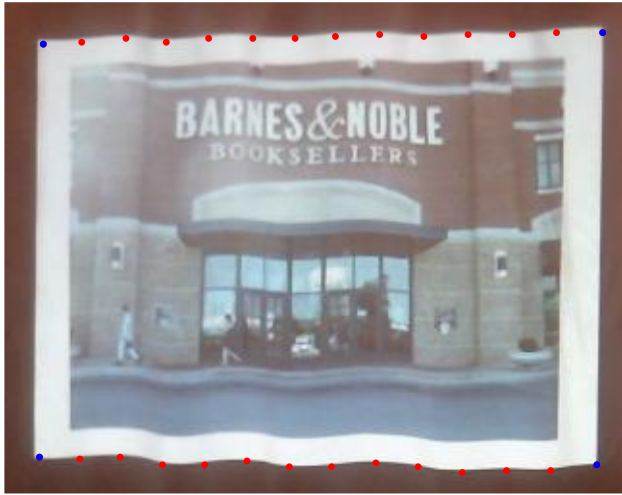
The results of various tests described in Section 4.5 performed to automatically solve the geometric distortion problem of images projected onto static non-planar display surfaces by using features of the natural image being projected are presented in the subsections following.

### **4.6.1 Edges and Corners at the Periphery of the Projected and Captured Images Used as Feature Points for Distortion Correction**

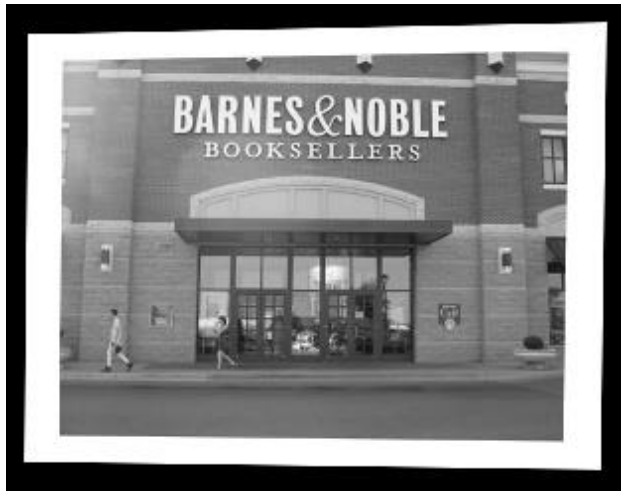
The results presented in figure 46 obtained from warping the entire image using a total of 28 control points derived from the top and bottom distorted edges of the image of the projection show that this distortion-correction procedure does not achieve the desired objective. Figure 46(e) shows that while distortion correction is achieved for the top and bottom edges, other features inside the image remain geometrically distorted. This is because RBFs are inherently local warping functions so that image points that are far away from control points along the periphery of the image do not get warped. The need to generate enough corresponding control points from important features inside both the projected image and the image of its projection in order to achieve distortion correction by warping the entire projected image is inevitable.



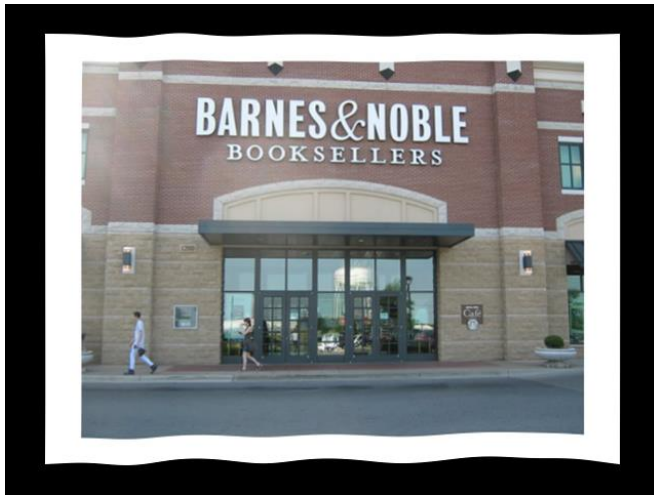
**(a) Non-distorted projected image**



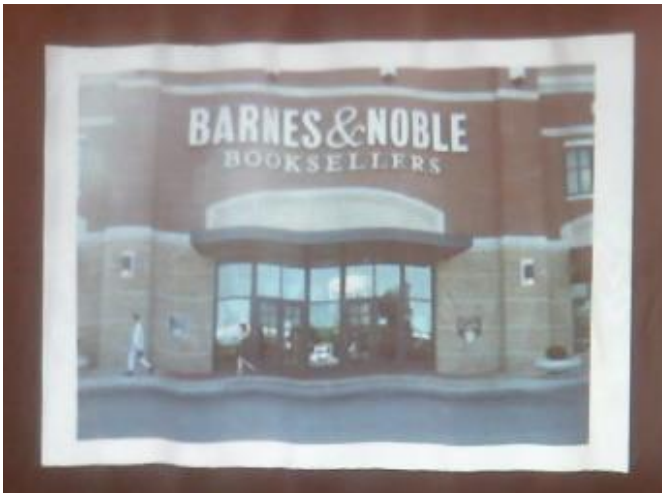
**(b) Distorted camera image of the projection of the image in (a) showing the 28 control points at the top and bottom edges (red and blue dots) and the image corner points (blue dots) that are linked to form the desired undistorted view of the projection (undistorted quadrilateral).**



**(c) The desired undistorted view of the projection**



**(d) The prewarped projected image showing that only its top and bottom edges undergo warping while points inside the image remain unaffected by warping**



**(e) The camera image of the projection of the prewarped input showing that distortion correction affects only the top and bottom edges of the image.**

**Figure 46: Results of auto geometric distortion correction by RBF warping using 28 control points from the top and bottom edges of the projected image**

#### **4.6.2 Distortion-correction Using Blob Features**

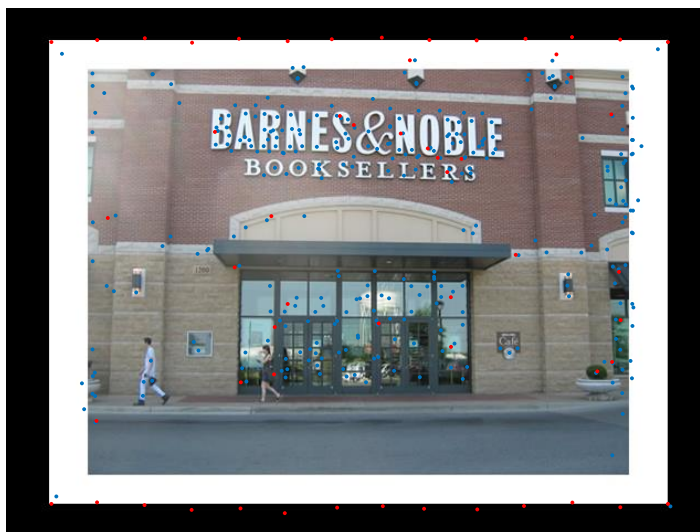
The results of geometric distortion correction by using blob features as control points obtained from the SURF implementation of the SIFT detection algorithm in addition to 28 other control points obtained from the periphery of the captured projection are presented in figure 47. The results show that this method still remains inadequate in solving the



geometric distortion problem because the arrangement of the resulting control points lack an identifiable geometric structure to which effective warping is based and the distribution of these points within the image is unpredictable and uncontrollable. Figure 47(a) illustrates the undesirable mismatch of blob features in two images and figures 47(b) and (c) illustrate the results of the process of eliminating unwanted control points after setting control point discrimination thresholds described in Section 4.5.2.



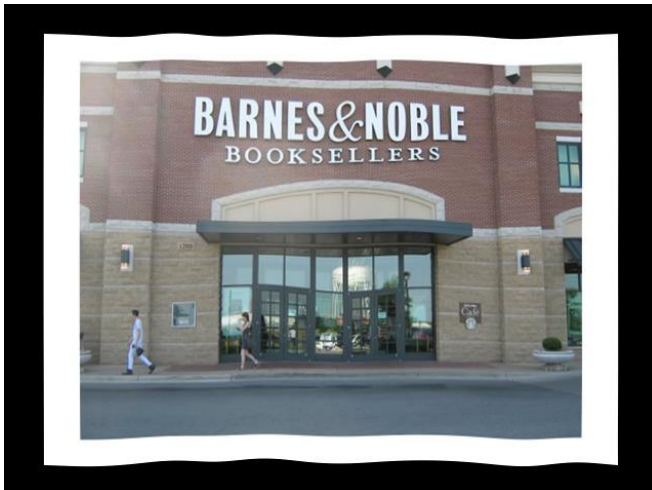
(a) The projected image and its captured display on the surface showing 29 correctly-matched sets of features and 1 incorrectly-matched set marked with an 'x'.



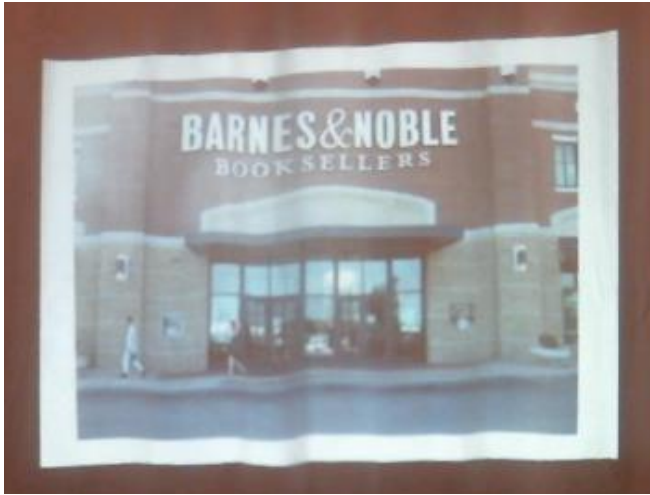
(b) The projected image showing all detected features in blue and red dots. Red dots represent the blob features remaining from the elimination process and additional control points obtained from the periphery of the image.



(c) Camera image of the projection showing all detected blob features in blue and red dots. Red dots represent the features remaining from the elimination process and additional control points obtained from the periphery of the image



(d) Pre-warped projected image. The figure shows that some regions inside the image have also been warped



**(e) Distortion-corrected image of the projection. Regions inside the image are observed to still remain geometrically distorted**

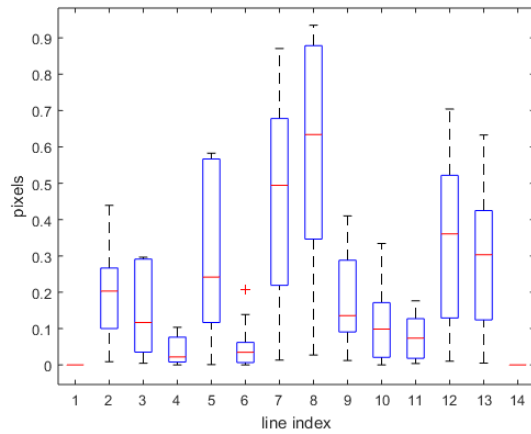
**Figure 47: Results of auto geometric distortion correction by RBF warping using blob features obtained from the SIFT algorithm.**

### **4.6.3 Linear Modelling of Distortion**

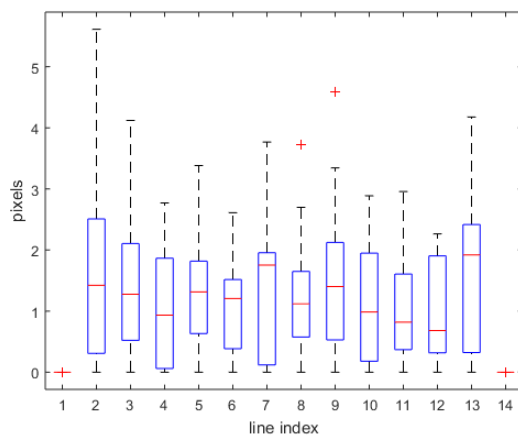
The results of distortion-correction by using a combination of feature points from the edges of the periphery of the image, its corners, and feature points inside the image obtained by linear interpolation of the horizontal and vertical displacements of extreme feature points with known displacement are presented in figures 48 through 51. Figures 48 (a) and (b) show box and whisker plots of the difference between horizontal and vertical pixel coordinates of feature points obtained by using the interpolation method to estimate 144 control points from 52 peripheral points. The plots show that an interpolation error of 1 pixel and less is obtained in the horizontal direction and less than 6 pixels in the vertical image direction. The average error in the vertical direction ranges between 0 and 2 pixels. The RBF coefficients used to warp the projected images shown in figures 49 (a) and (b) were obtained by using a calibration image and linear interpolation respectively. Both images show very identical warping trends for the two methods. The latter figure also shows that internal control points obtained by linear interpolation produced satisfactory warping. The similarity between both images is also revealed by a peak NCC value of 0.9548.

The quality of the observed outputs obtained from both methods is shown in figures 50(a), (b) and (c). The illustration in figure 50 (c) obtained from linear interpolation shows appreciable correction of the observed distorted projected image shown in figure 50(a) and close similarity to the distortion-corrected projection in figure 50(b) obtained from using a calibration image. The peak NCC value obtained by matching the desired observed projection with the distortion-corrected projection obtained by linear interpolation of control point displacement was 0.9368, an improvement from 0.8898 obtained from matching the observed distorted projection with the desired observed projection.

The NCC matching between observed projections from both methods of distortion correction gave a good peak value of 0.9921. Figure 51 shows the improvement of peak NCC values for distortion correction for 10 different projector/camera poses and projection surface shapes.

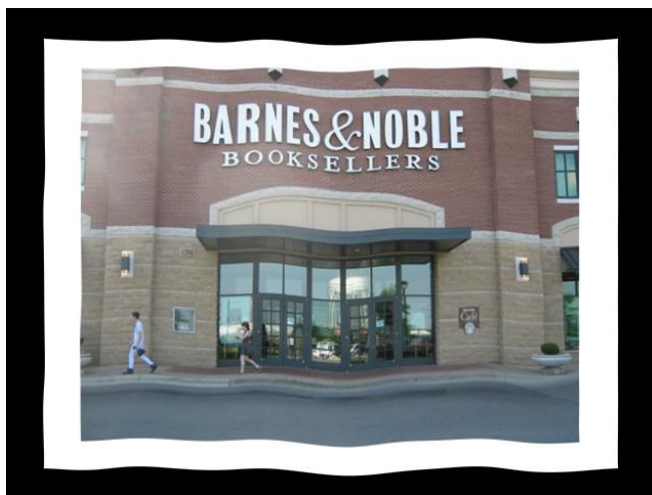


**(a) Absolute differences for horizontal pixel coordinates**

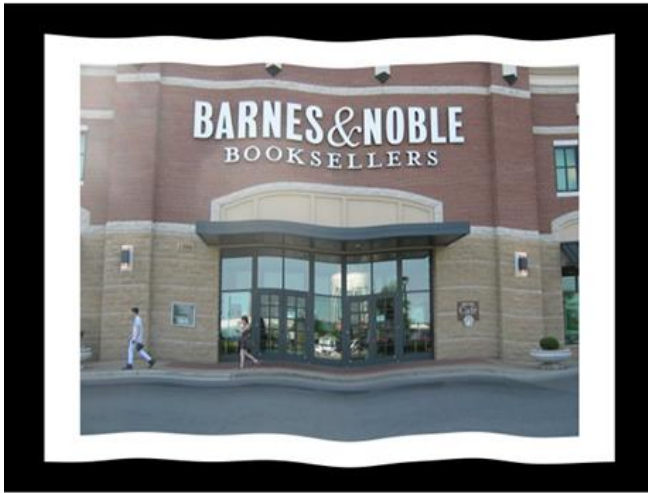


**(b) Absolute differences for vertical pixel coordinates**

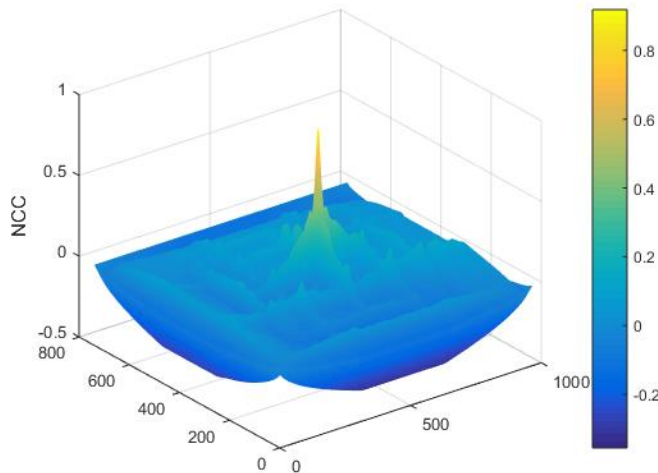
**Figure 48: Box and whisker plots for a single image showing the absolute difference between horizontal and vertical pixel coordinates obtained from using a calibration image and coordinates obtained from linear interpolation of coordinate displacements.**



**(a) Warp obtained from the calibration image**

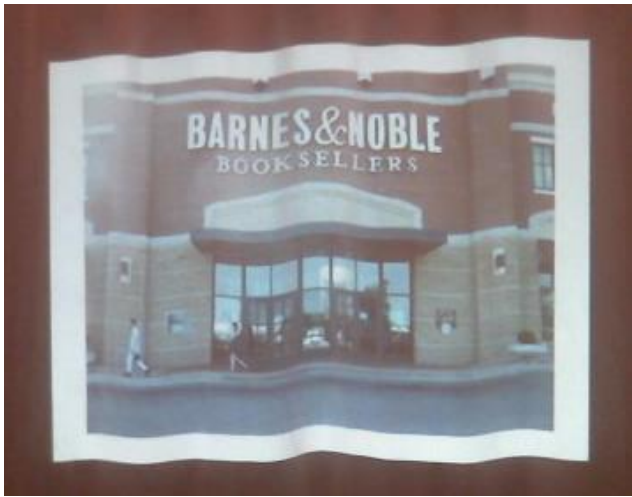


**(b) Warp obtained from linear estimation of control point displacements**

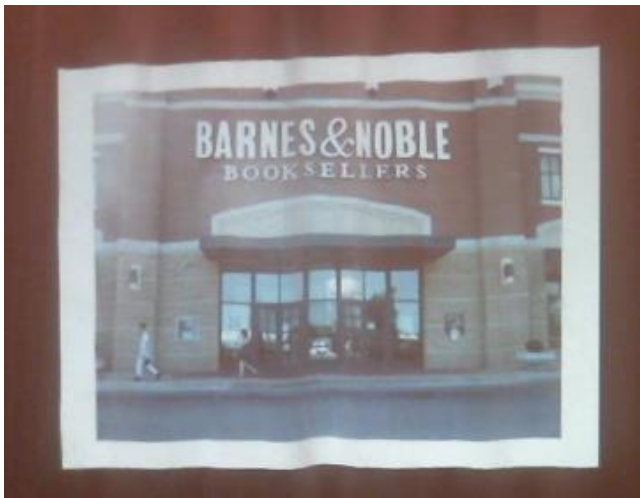


**(c) Plot of NCC matching values between images in (a) and (b). Peak NCC = 0.9518.**

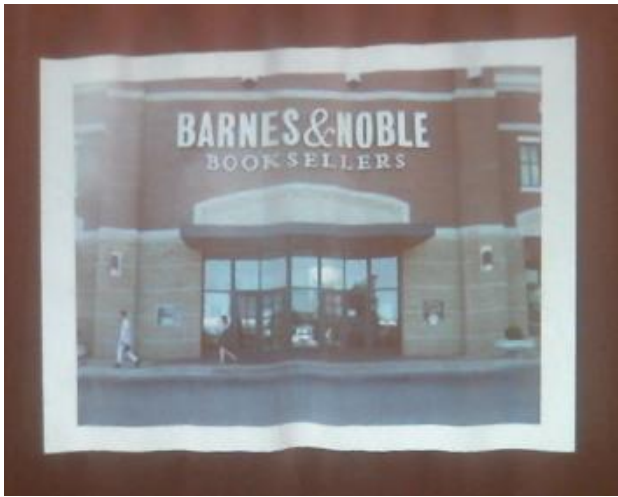
**Figure 49: Comparison between pre-warped projected images by RBF coefficients obtained from using a calibration image (a) and RBF coefficients obtained from linear interpolation of deviation of control points on the periphery of the projected image (b), with the surface plot of NCC matching values for both images (c).**



**(a) Observed geometrically-distorted projection**



**(b) Observed distortion-corrected projection using a 196 point calibration image**



(c) Observed distortion-corrected projection using linear interpolation of displacement of control points. Peak NCC between (b) and (c) is 0.9921.

Figure 50: Observed geometrically distorted projection (a) and distortion-corrected projections for methods involving the use of the 196-point calibration image (b) and that obtained from using 52 known control point displacements and 144 linear interpolated control point displacements.

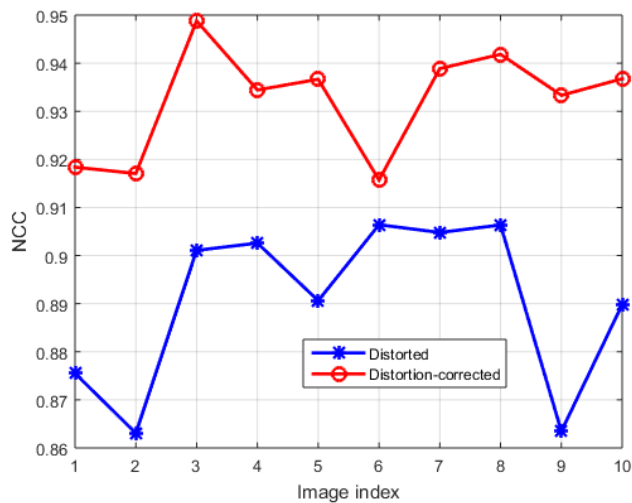


Figure 51: Improvement of peak NCC matching values between the desired observed projection and the observed distorted projection and distortion-correction by linear interpolation of control point displacement for 10 different poses of the projector, camera and shapes of the projection surface.



## 4.7 Chapter Summary

The main focus of this chapter has been the correction of geometric distortion for projections on static non-planar displays or surfaces. The highlights drawn from the discussions in this chapter include the following:

(i) The development of a simple pictorial and algebraic model of a straight line projected onto a cylinder used to illustrate the geometric distortion that occurs when non-distorted shapes are projected onto non-planar surfaces. It was shown that in order to observe a non-distorted projection on the quadric surface, projected lines have to be warped in directions that oppose the directions of the observed distortion.

(ii) The use of a projected calibration image consisting of a grid of feature (corner) points to measure the amount of distortion suffered by projected horizontal and vertical lines on the projection surface. The distortion of these lines was measured by calculating the average perpendicular distance between points on the observed distorted lines and the desired observed straight lines.

(iii) Comparing distortion histograms of observed distorted horizontal and vertical lines of the projected calibration image with distortion histograms of a random selection of standard straight image lines showed the horizontal lines were significantly more distorted than both the vertical lines and standard straight image lines. The vertical lines were found to be very close (within 1 pixel) in distortion measures to the standard straight lines.

(iv) The first technique developed to correct distortion for static non-planar surfaces relied on projecting the calibration image to measure the observed distortion of feature points from their respective lines. The features of the distorted camera-captured view of the projected calibration image were used to work out the warping trend of the projected calibration image through straight line geometry and 2D homography transformations. The entire calibration image was warped using RBF coefficients obtained from the coordinates of its initial non-distorted control points and its final warped control points.

The same coefficients were used to warp an entire projected natural image to correct the distortion of its observed projection while the position of components of the projection system remained static.

(v) The peak Normalised Cross-correlation Coefficient (NCC) obtained from matching respective camera images of the distorted and distortion-corrected projections with the standard non-distorted homography-shaped image was used as the measure to indicate the effectiveness of the distortion-correction technique developed.

(vi) Distortion corrections with calibration images having 144 and 196 control points yielded better correction results with peak NCC values improved as much as 31% than calibration images with 36, 64, and 100 control points revealing that distortion-correction is achieved with calibration images having a sufficient number and distribution of control points. A guide to determining the number of control points to use is having at least 1 control point per point of inflexion of a crest or trough per distorted line.

(vii) The three RBFs compared yielded similar distortion-correction results but the multiquadric RBF was chosen over the other two (inverse multiquadric and Gaussian) because of its shorter operation time of 19.97s.

(viii) The distortion-correction technique was also successful on the non-planar projection surface formed by two intersecting planar walls as it was on a freely-hanging curtain.

(ix) An effective automatic technique used to correct distortion for static non-planar surfaces relies on estimating the displacement of gridded control points from their non-distorted positions by applying a linear interpolating model between the observed displacement of the control points on the peripheral edges of the projected image. It eliminates the need to project a calibration image and performs the distortion-correction task better than using natural image features from the SIFT technique for blob detection and using only feature points on the periphery of the projected image.

(x) Peak NCC values as high as 0.95 were obtained from matching warped projected images whose displaced control points were obtained by the linear modelling technique with warped projected images whose displaced control points were obtained by projecting the calibration image. Peak NCC values as high as 0.99 were also obtained from matching observed distortion-corrected projections from both techniques.

# CHAPTER FIVE

## Geometric Distortion Correction for Deformable and Dynamic Projection Systems

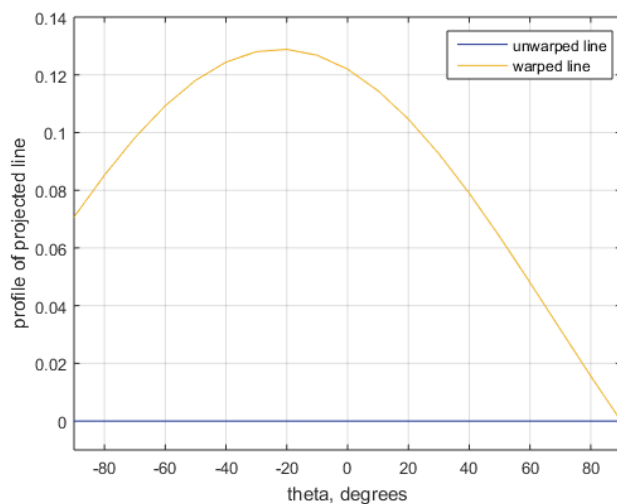
### 5.1 Introduction

This chapter deals with techniques developed to solve geometric distortion for non-planar projection systems that are deformable and/or dynamic.

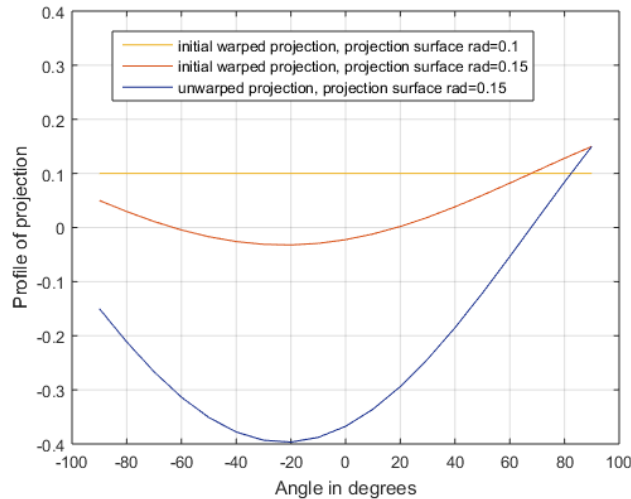
Dynamic projection systems are characterised by components of the system changing with time. These include a projector and camera whose positions change with respect to the projection surface or a projection surface whose position changes with time. The shape of a deformable projection surface could also change with time. The discussion in Section 5.2 begins with the illustration similar to that given in Section 4.2 of an upright cylindrical projection surface whose shape changes while the state of a previous projection remains the same. It demonstrates a method of warping that can be generalised for typical systems and applied in experiments described in Section 5.3 and the results presented in Section 5.4. An example of a typical dynamic projection system is one utilised in simulated golf where the player hits the ball against a non-planar deformable surface. The impact of the golf ball on the surface causes the image and text being projected onto the surface to become distorted. The correction of this type of distortion will therefore require the distortion-correction system to work in real-time. Code profiling and optimisation for real-time performance is treated in Sections 5.5 and 5.6. Experiments involving striking the projection surface with a projectile and correcting distortion in real-time are presented in Section 5.7 and results of these are presented in Section 5.8. The control points of projected images warped in this chapter were obtained by the method of linear modelling of distortion developed in Section 4.5.3. A summary of the main discussions and findings in this chapter is presented in Section 5.9.

## 5.2 Illustrating Geometric Distortion of a Warped Image Line Projected onto a Deformable or Dynamic Quadric Surface

Following the algebraic illustration of geometric distortion of the projection of a line on a cylinder given in Section 4.2.2, the reintroduction of a distorted view of the projection by changing the shape of the surface is shown in figure 52. To begin with, the projection of a straight line onto the cylinder produces a distorted view on the cylinder's surface. The correction of this distorted view involves warping the line to a shape that is opposite to that of the observed distortion. The orange curve in figure 52(a) has been warped from its original shape (blue curve) to the orange curve in the same figure to produce the non-distorted view (orange curve) in figure 52(b). The observed projection becomes distorted again with the radius of the cylinder changing from  $r_1 = 0.1$  to  $r_2 = 0.15$  as shown by the brown curve in figure 52(b). The shape of this distorted curve is significantly different from that observed when a straight line is projected onto the surface of the cylinder of radius  $r_2$  as shown by the blue curve in figure 52(b). Two different strategies may be used to correct the distortion reintroduced by changing the shape of the projection surface: The first strategy involves warping the already-warped line to its new shape while the second involves warping the straight line to the same new shape. The re-warping strategies applied to a real dynamic projection system and their advantages and disadvantages are discussed in the following Section 5.3.



(a) Profile curves of the warped and unwarped projected line



**(b) Profile curves of the observed warped and unwarped projected line before and after the change of shape of the projection surface**

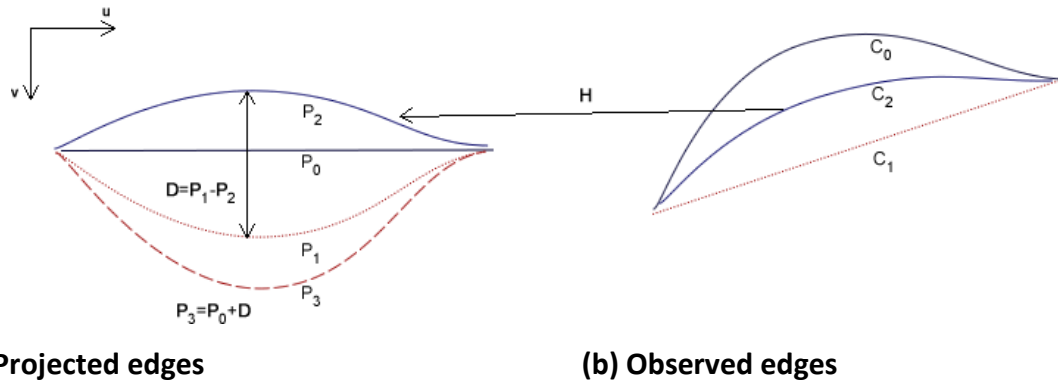
**Figure 52: Curves showing the profiles of the warped and unwarped projected lines and their respective profiles on the quadric projection surface before and after the change of shape of the projection surface**

### 5.3 Distortion Correction for Real Dynamic Projection Systems

The distortion-correction technique used when the projector and/or the camera and/or the shape of the projection surface change after the projected image is warped to a new state is described in this section. The images of the projection captured by the camera for the dynamic projection system have to be treated differently than those of a static system because these distorted images are obtained from an already-warped projected image rather than an unwarped one as is the case of a static projection system.

An illustration of this dynamic case is given in figure 53. According to figure 53, let the edges of the projected and observed images be denoted by  $P$  and  $C$ , and the coordinates of points on these edges be  $\mathbf{P}$  and  $\mathbf{C}$  respectively. For convenience the horizontal and vertical image coordinates increase in the rightward and downward directions respectively.

(i) Suppose the unwarped projected edge  $P_0$  is observed as distorted edge  $C_0$ .  $P_0$  will then be warped to  $P_1$  in order to observe the distortion-corrected edge  $C_1$ .



**Figure 53: Distortion-correction image warping strategy for dynamic projection systems**

(ii) Let a change in the projection system cause  $C_1$  to be observed as distorted edge  $C_2$ .  $C_2$  transforms to  $P_2$  by the homography  $H$  between the observed image and the projected image. To correct for this new distortion, the projected edge should be warped to  $P_3$  from  $P_0$ .

(iii) In determining the coordinates of  $P_3$  the amount of distortion-causing deviation  $D$  that accounts for the difference between the distortion-corrected view and the distorted view in the projected image domain is given as

$$D = P_1 - P_2 \quad (60)$$

where  $P_1$  and  $P_2$  represent the coordinates of the points on edges  $P_1$  and  $P_2$  respectively.

(iv)  $D$  is added to  $P_0$  to obtain  $P_3$ , the coordinates of points on the edge  $P_3$ .

$$P_3 = P_0 + D$$

$$\text{or } P_3 = P_0 + P_1 - P_2 \quad (61)$$

Equation 61 implies that the coordinates of the present warp must be memorised in order to be available to use in the next warp for a continuously-changing projection system.

Also according to the analysis of the system given, two re-warping strategies of the projected image exist to correct the distortion and they include:

(i) Re-warping the already-warped image: This is equivalent to warping the projected image from already-warped position  $P_1$  to  $P_3$  in figure 53.

This strategy works well when applied to the same image once or twice. Beyond this number of corrections the quality of observed projection deteriorates because of cumulative errors introduced by successive RBF and bilinear interpolations applied to the image. It will also not be possible to apply this strategy for correcting the distortion of projected video, because the source video frames are always played back undistorted with only the unwarped control points known. Video frames therefore need to be pre-warped from their original undistorted position.

(ii) Warping the unwarped projected image each time the projection system changes: This is equivalent to warping the projected image from position  $P_0$  to  $P_3$  in figure 53. This method prevents the build-up of geometric errors from one image frame to the next. The results of distortion correction using the two dynamic projection system image re-warping strategies described above are given in Section 5.4 following. The two distortion-correction processes carried out here were done with the automatic warping technique based on linear modelling of distortion described in Section 4.5.3

#### **5.4 Results of Tests Carried Out for Distortion-correction Techniques on Real Dynamic Projection Systems**

The set of results shown in figures 54 through 57 were obtained by successively correcting the distortion and changing the shape of the projection surface and/or changing the pose of the camera or projector to re-distort the observed projection. The results for the two re-warping strategies described in Section 5.3 support the claims that warping the projected image each time from its undistorted state performs better than warping it from its already-warped state with observed straighter edges and higher peak NCC values. Figure 54 shows the resulting observed projections after warping the projected image from its already-warped state upon each dynamic projection system state change. Figure 55 shows a plot of peak NCC values obtained from one dynamic projection state change to another. The graph reveals that distortion-correction by warping the projected image from its already warped state sometimes yields worse observed distortions with lower peak NCC values than the previously-observed distorted projections.



Figure 56 shows the resulting observed projections after warping the projected image from its unwarped state upon each dynamic projection system state change. The accompanying figure 57 shows a plot of the peak NCC values obtained for each dynamic projection state after distortion-correction. The graph reveals that distortion-correction by warping the projected image from its unwarped state after every dynamic projection state change always yields distortion-corrected projections.



**(a, i) Undistorted projected image**



**(a, ii) Observed distorted projection (I),  
peak NCC = 0.8885**



**(b,i) Pre-warped projected image (I)**



**(b,ii) Observed distortion-corrected  
projection (I) peak NCC = 0.9213**



**(c,i) Pre-warped projected image (I)**



**(c,ii) Observed distorted projection (II)**

**peak NCC = 0.9038**



**(d,i) Pre-warped projected image (II)**



**(d,ii) Observed distortion-corrected projection (II)**

**peak NCC = 0.9025**



**(e,i) Pre-warped projected image (II)**



**(e,ii) Observed distorted projection (III),**

**peak NCC = 0.8904**



**(f,i) Pre-warped projected Image (III)**



**(f,ii) Observed distortion-corrected Projection (III), peak NCC = 0.8991**



**(g,i) Pre-warped projected Image (III)**



**(g,ii) Observed distorted projection (IV), peak NCC = 0.9097**

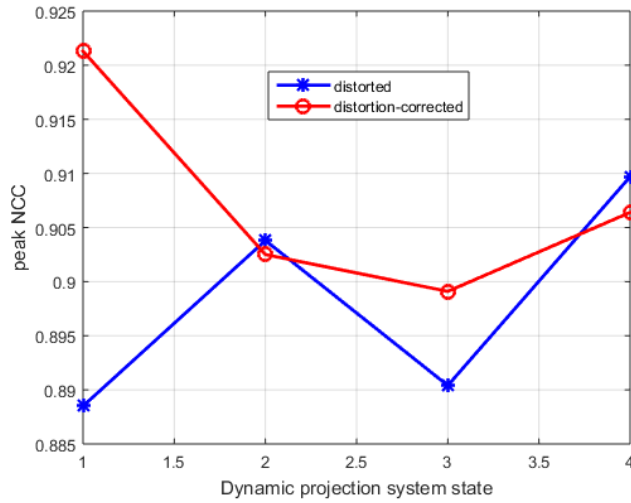


**(h,i) Pre-warped projected image (IV)**



**(h,ii) Observed distortion-corrected projection (V), peak NCC = 0.9064**

**Figure 54: Results for successively correcting, distorting, and correcting the observed projection by warping the projected image from its previously-warped state**



**Figure 55: Graph showing peak NCC values for successively correcting, distorting, and correcting the observed projection by warping the projected image from its previously-warped state.**



**(a, i) Undistorted projected image**



**(a, ii) Observed distorted projection (I),  
peak NCC = 0.8693**



**(b, i) Pre-warped projected image (I)**



**(b,ii) Observed distortion-corrected  
projection (I), peak NCC = 0.9292**



**(c, i) Pre-warped projected image (I)**



**(c, ii) Observed distorted projection (II),  
peak NCC = 0.9014**



**(d, i) Pre-warped projected image (II)**



**(d,ii) Observed distortion-corrected  
projection (II), peak NCC = 0.9245**



**(e, i) Pre-warped projected image (II)**



**(e, ii) Observed distorted projection (II),  
peak NCC = 0.9082**



**(f, i) Pre-warped projected image (III)**



**(f, ii) Observed distortion-corrected projection (III), peak NCC = 0.9291**



**(g, i) Pre-warped projected image (III)**



**(g,ii) Observed distorted projection (III), peak NCC = 0.8944**

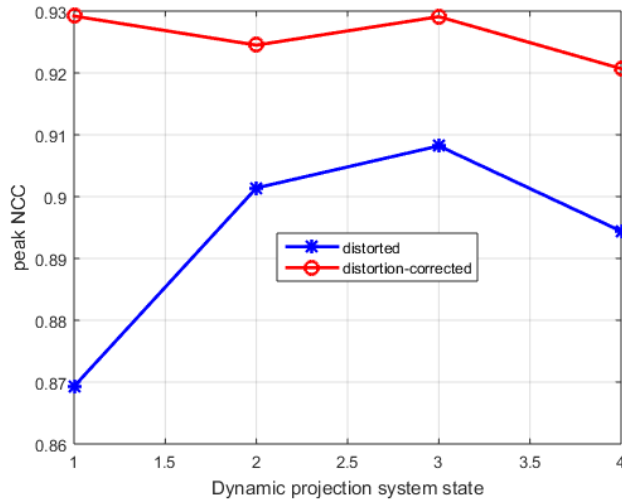


**(h,i) Pre-warped projected image (IV)**



**(h,ii) Observed distortion-corrected projection (IV) peak NCC = 0.9207**

**Figure 56: Results for successively correcting, distorting, and correcting the observed projection by warping the projected image from its unwarped state**



**Figure 57: Graph of peak NCC values obtained from successively correcting, distorting, and correcting the observed projection by warping the projected image from its unwarped state.**

### 5.5 Code Profiling and Optimisation towards Real-time Operation

For the distortion-correction technique designed in this work to correct every frame of observing video of a continuously-changing projection system designed to operate at 15fps (frames per second), the processing time required for each frame should be less than 0.06s. Table 5 lists the eight (groups of) operations necessary to correct distortion for dynamic projection surfaces and the average time each operation took to complete on the processing platform used. For the whole distortion-correction process to complete within the target time of 0.06s each operation has to complete in less than 0.00375s on the average including overheads. The table shows that no one operation meets the target as the current capability of the distortion-correction system stands at 0.047fps.

Necessary time-saving changes must however be made to individual operations to improve performance of the overall system to meet the target framerate. Possible time-saving changes to the operations shaded in table 5 were carried out in this work. The strategies employed are presented in the following subsections. Results of the changes are presented in Section 5.6.

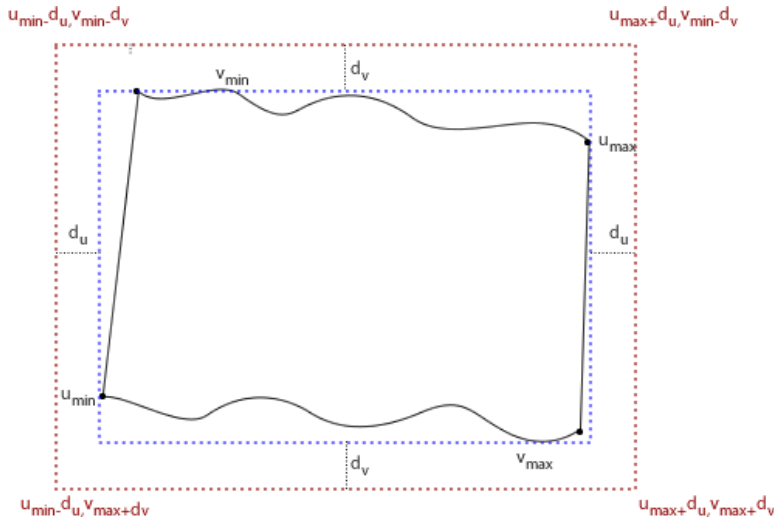
**Table 5: Average time taken for critical image distortion-correction operations to complete on Matlab**

Image	Description of group of operations	Total time (s)	Description of operation	Time taken to run (s)	% Total time taken to run
<b>CAMERA-OBSERVED IMAGE</b>	<i>Detection of 52 feature points, Homography estimation and observed-to-projected image points transformation</i>	0.3614	Detection and arrangement of 4 corner points	0.2112	0.99%
			Detection and arrangement of 48 other edge points	0.1371	0.64%
			Estimation of projected-observed image homographies and transformation of 52 feature points from the observed image to the projected image	0.0131	0.06%
<b>PROJECTED IMAGE</b>	<i>Estimate Control points, RBF coefficients and warp RGB image</i>	20.9906	Estimation of 144 other control points from the previous 52 to produce 196 control points	0.0006	0.003%
			Calculation of 398 RBF coefficients from 196 control points	0.0289	0.14%
			Calculation of the warped coordinates of the projected image	20.6773	96.84%
			Bilinear interpolation of warped image values: R,G and B components	0.2752	1.29%
			Updating projected image display on computer	0.0086	0.04%

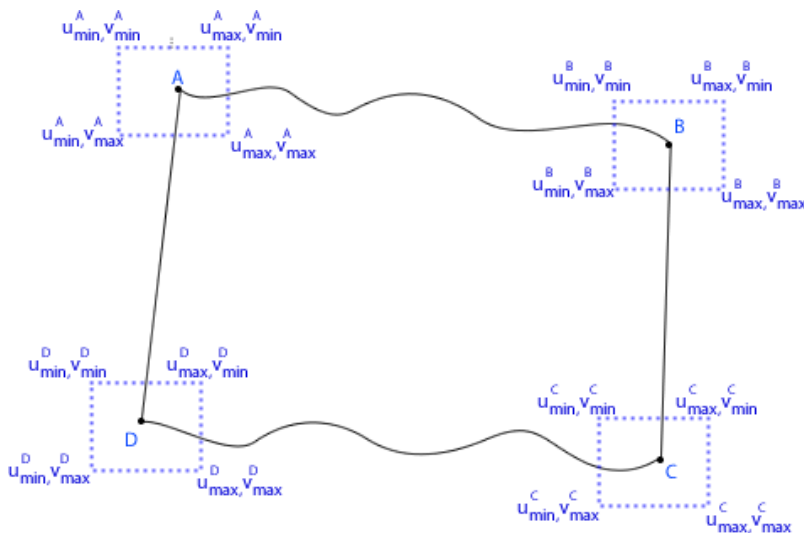


### 5.5.1 Creating a Region-of-Importance (ROI) for Edge and Corner Detection

Defining an ROI for the image or video being projected reduces processing time because the ROI presents a smaller region of pixels for processing than the entire image does. Time savings may however be hampered by the time required to derive and apply the ROI in the processing. The figures 58(a) and (b) show illustrations of ROIs developed for quicker detection of edges and corners.



(a) ROI for edge detection



(b) ROI for corner detection

Figure 58: Illustrating the principle of developing ROIs for faster edge and corner detection

For a particular application, for example when striking the projection surface with a projectile, a suitable ROI for edge and corner detection can be created with knowledge of the expected maximum deviation that the edges and corners of the observed projection suffer in both image directions after the projectile strikes the surface. Figure 59 shows observed projected images before and after the projection surface was hit by a projectile.



**(a) Observed projection before a projectile strike**



**(b) Observed projection after a projectile strike**

**Figure 59: Observed consecutive frames of a projection on a surface hit by a projectile in an experiment to investigate the deviation of feature points from one frame to the next on a deforming projection surface.**

The following terms were defined for the development of an ROI for edge detection as shown in figure 58.

$u_{min}$  = Smallest horizontal coordinate value of all edge points lying on the periphery of the image

$v_{min}$  = Smallest vertical coordinate value of all edge points lying on the periphery of the image.

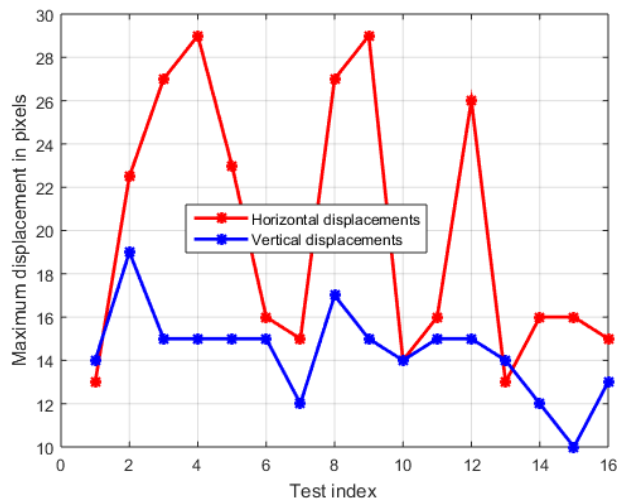
$u_{max}$  = Largest horizontal coordinate value of all edge points lying on the periphery of the image.

$v_{max}$  = Largest vertical coordinate value of edge all edge points lying on the periphery of the image

$u_{min}, v_{min}, u_{max}, v_{max}$  were obtained from the previous frame

$d_u$  = A value subtracted from  $u_{min}$  and added to  $u_{max}$  to account for the amount of displacement suffered in the horizontal direction in the present frame

$d_v$  = A value subtracted from  $v_{min}$  and added to  $v_{max}$  to account for the amount of displacement suffered in the vertical direction in the present frame. The values of  $d_u$  and  $d_v$  are influenced by the estimated maximum displacements suffered by the distorted image in the respective horizontal and vertical directions. They are application-dependent. In this work tests to determine  $d_u$  and  $d_v$  were carried out by striking the projection surface with a squeezed paper projectile and recording the respective absolute maximum displacements of feature points from frame to frame. Figure 61 shows a plot of absolute maximum displacements for 16 tests. Each test consisted of a minimum of 27 video frames. The figure shows that the maximum displacement recorded in all tests was 29 pixels horizontally and 19 pixels vertically. Intuitively 30% of the greatest absolute displacements in each direction was added to or subtracted from the respective displacements to arrive at values for  $d_u$  and  $d_v$  to make the ROI for edge detection span from  $\max(1, (u_{min} - d_u))$  to  $\min(640, (u_{min} + d_u))$  horizontally and  $\max(1, (v_{min} - d_v))$  to  $\min(480, (v_{min} + d_v))$  vertically to ensure the entire ROI is contained within the image.



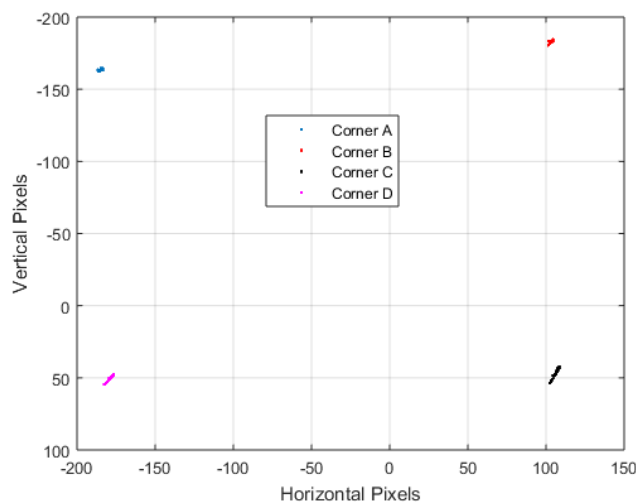
**Figure 60: Observed displacement of edge pixels in the horizontal and vertical image directions after striking projection surface with a projectile**

The same  $d_u$  and  $d_v$  values obtained for edge detection were used to determine an ROI for each of the four corners of the image. This was done by adding the values of the estimated maximum displacement in each direction to the actual pixel coordinates of each corner from the previous frame. ROI spanned from  $\max(1, (u_x - d_u))$  to  $\min(640, (u_x + d_u))$  horizontally and from  $\max(1, (v_x - d_v))$  to  $\min(480, (v_x + d_v))$  vertically where  $u_x$  and

$v_x$  represent the horizontal and vertical pixel coordinates of each corner respectively. The results in Table 6 in Section 5.6 show the factors by which the application of ROIs for edge detection and corner detection shortened the calculation of feature points. To improve on the performance of this method, two separate but smaller ROIs can be created for edge detection and the detection of corners by a corner detection algorithm may be entirely avoided. This method is presented in Section 5.5.2 following.

### 5.5.2 Estimating Corners from Edge Information

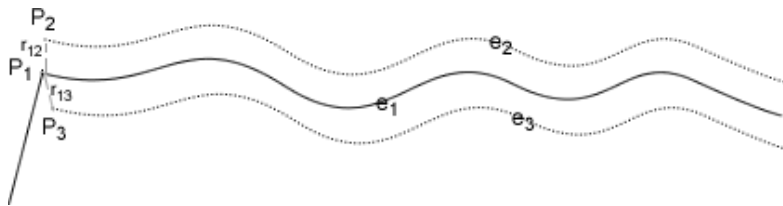
Because the detection of corners takes a considerably long time of about 0.21s to complete, the edge detection can be performed and the four needed corners estimated from the detected edge pixels as a corner is formed where two distinct edges going in different directions meet. It is observed that when the projectile strikes the projection surface the edges of the distorted image are not displaced too many pixels from their mean position as shown in figure 61. Because these corners also form part of the detected edges, their displacements from the mean positions may be estimated by determining the shortest distances of detected edges from pre-determined mean corner pixel positions.



**Figure 61: Two-dimensional variation of the four corners of observed projected video frames on a deforming surface.**

Consider the corner in figure 62 at its mean position  $P_1(u, v)$  is part of the edge  $e_1$  also shown in its mean position. It is assumed that the corner will always be detected with the

edge whenever the edge changes position to  $e_2$  or  $e_3$  for example as long as the edge does not wrap around itself such as to occlude the view of the corner. The corner is also displaced to positions  $P_2$  and  $P_3$  as the edge moves to  $e_2$  and  $e_3$  respectively. From the illustration, it can be seen that the distance  $r_{12}$  from  $P_1$  to its displaced position  $P_2$  will be the shortest between the corner and any point on the edge, assuming the corner and edge do not suffer too large a displacement. The same shortest-distance principle applies to  $r_{12}$  and  $e_3$ . The positions of displaced corners in frames of an observed projection can therefore be estimated by calculating the distances between the points on the displaced edge and the mean position of the corner. The position of the corner will be that point with the shortest distance from the mean position of the corner. In this work the mean positions of the corners are taken as those of the corners at rest in the first frame. Therefore the corner detection is carried out only once and the prediction is taken from a single measurement. This method therefore replaces longer gradient-finding computations involved in corner detection with much fewer multiplications and additions. Table 6 shows the closeness between the coordinates of corner points in the first frame of a video and their actual means obtained throughout the duration of the video in a test to measure the displacement of edge and corner points of a projected image when the projection surface was struck by a projectile.



**Figure 62: Illustration of how to estimate corners from edges**

**Table 6: Comparison between estimated mean coordinates of corners obtained from the first frame of a video and their actual mean coordinates obtained from the whole video duration of 24 frames**

Corner	A	B	C	D
<i>u coordinate first frame</i>	135.0	422.0	425.0	141.0
<i>Mean u coordinate</i>	135.2	422.5	425.8	141.5
<i>u coordinate estimation error</i>	0.14%	0.12%	0.19%	0.35%
<i>(u) coordinate range</i>	3.0	3.0	6.0	6.0
<i>v coordinate first frame</i>	76.0	58.0	287.0	289.0
<i>Mean v coordinate value</i>	75.7	57.4	286.0	288.9
<i>v coordinate estimation error</i>	0.4%	1.04%	0.35%	0.03%
<i>v coordinate range</i>	2.0	4.0	11.0	7.0

A method to test the closeness of the estimated positions of the corners to their true positions is to compare the respective magnitudes of the estimated and true corners and the angle between them when they are represented as vectors. Equations 62 and 63 define the desired closeness measure

$$|p| - |c| = 0 \quad (62)$$

$$\cos \theta = \frac{p \cdot c}{|p||c|} = 1 \quad (63)$$

Where  $p$  and  $c$  are the estimated and true position vectors of the corners respectively and  $\theta$  is the angle between them. The results of tests to determine the amount of time savings of computations using this method and the accuracy of estimating corner points with information from edges are presented in Section 5.6.2.

### 5.5.3 Using Linear Interpolation to Approximate the Warping of the Projected Image

The extremely long time used to calculate the new pixel coordinates by directly evaluating equations 45 and 46 can be considerably shortened by applying linear interpolations similar to that used in the estimation of internal control points of the projected image as described in Section 4.5.3. In this case the equations are fully evaluated for the top and bottom rows

of the image. All other warped coordinates between them are estimated using linear interpolations to obtain the new pixel positions for the rest of the image.

Results for tests obtained by employing this method to evaluate the coordinates of the warped projected image are presented in Section 5.6.3.

#### **5.5.4 Greyscale versus RGB Processing**

When warping an RGB image, the grey values of each component in the warped image have to be calculated through bilinear-interpolation separately so that the total time used in processing the RGB image is about 3 times the time required for processing a greyscale image with only one component. Projecting and processing a greyscale image therefore reduces the bilinear interpolation processing required for a hole-free warping by a factor of 3. Greyscale projection and processing is only considered as a potential processing-time reduction measure because of the quality of the observed projection that could ultimately affect the viewer's satisfaction. Results are shown in code-optimisation summary Table 10.

#### **5.5.5 Parallel Processing**

This option parallelises routines that can be performed in parallel using the resources of the computer and compatible programming language and processor. The Graphics Processing Unit (GPU) having up to thousands of processors on one chip, or the main processor on a PC having much fewer processors than the GPU can be used for parallel processing. Parallel processing operations with the GPU are possible in Matlab only with GPUs made by NVIDIA. Unfortunately only high-end PCs so far have the NVIDIA GPUs. Parallelising operations on a standard PC can be done by coding with multithreading-capable programming languages like C, C# and C++. Parallelising routines is also only possible for routines that have individual mutually-exclusive operations. Routines designed for gridded data in this work have individual operations that can be parallelised. These routines include (i) edge and corner detection of the projected image using clearly-defined ROIs, (ii) camera image-to-projection image homography transformations, (iii) calculation of RBF warping coefficients, (iv) calculating the warped coordinates of the projected image, (v) bilinear interpolation of grey values of the warped projection image.

## 5.6 Results of Code Profiling and Optimisation for Real-time Performance

The results presented in this section follow the order in which they are treated in Section 5.5. In each case only the relevant operations from Table 5 and their respective timings appear in the tables that follow. There are negligible differences in timings of the same operations in Table 7 and Table 8 compared with Table 5 because each operation differs in time from one run of the operation to another possibly due to the computer's operating system-controlled scheduling of processes.

### 5.6.1 Code Profiling Results Obtained for Creating an ROI for Edge Detection and 4 ROIs for Corner Detection.

Table 7 shows that creating ROIs for edge and corner detection reduced the times required for the detection of feature points by more than 45%. The times recorded for the detection of features with the use of ROIs include the time spent creating the ROIs in order of around  $10^{-4}$ s.

**Table 7: Processing time saving resulting from creating ROIs for edge and corner detection**

Operation	Time to complete (s)		
	Without ROI (1)	With edge & corner ROI (2)	% Reduction $= \frac{(1)-(2)}{(1)} \times 100$
Detection of 52 feature points, Homography estimation and transformation	0.3588	0.1946	45.8%



### 5.6.2 Code Profiling Results Obtained from Estimating Four Corners from Detected Edges.

**Table 8: Processing time saving resulting from creating 2 ROIs for edge detection and estimating corners from detected edges**

Operation	Time to complete (s)			
	Without ROI (1)	With edge & corner ROIs (2)	With 2 edge ROIs & corner estimation (3)	% Reduction $= \frac{(1)-(3)}{(1)} \times 100$
Detection of 52 feature points, Homography estimation and transformation	0.3588	0.1946	0.0524	85.40%

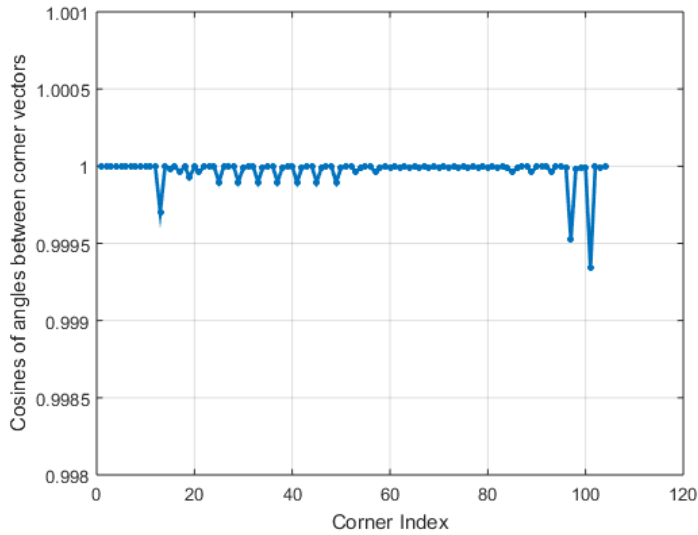
Table 8 shows that the estimation of corner points from detected edges using ROIs defined for the top and bottom edges is the best option for time saving by yielding a time saving of 85.4%.

Figure 63 shows two frames from a test with a projectile hitting the projection surface. The estimated corner points shown in red dots are seen to be very close to the actual corner points obtained from the Harris corner detector marked with blue dots.

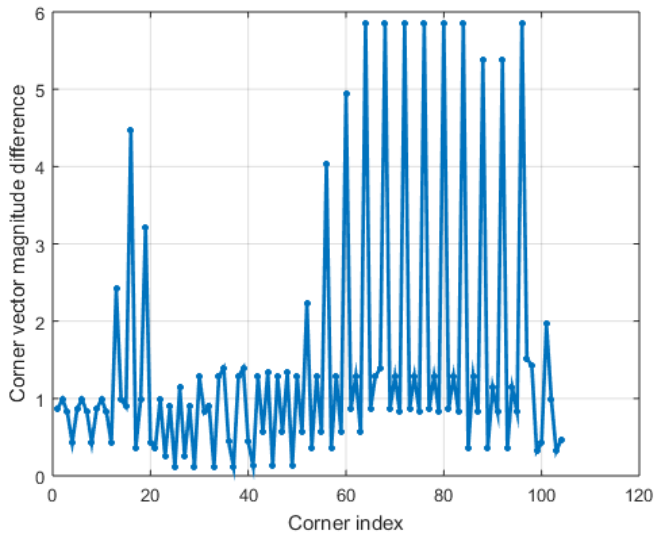
Figure 64 shows the absolute difference in magnitudes of the estimated and true corner vectors and cosine of angles of separation between them obtained from a projectile test with 26 frames. They show a good estimation with maximum absolute magnitude differences of maximum of 5.8 pixels and maximum angle of 1.8°.



**Figure 63: Pictorial illustrations to show the closeness between estimated corners and actual corners. Estimated corners are red dots while actual corners are blue dots.**



**(a) Cosines of angles between estimated corner point vectors and true corner point vectors.**



**(b) Absolute differences in magnitudes between estimated corner point vectors and true corner point vectors**

**Figure 64: Comparison between values of estimated and true corner points using cosines of angles and absolute magnitude differences between their respective vectors.**

### 5.6.3 Code Profiling Results Obtained from Using Linear Interpolation to Approximate the Warping of the Projected Image

Two sets of results are presented in this section. The first result in Table 9 shows that using linear interpolation to approximate the warping of the projected image having calculated the warped coordinates of the top and bottom edges using the warping function in equations 45 and 46 reduces the operation time drastically by about 98.8%.

More time saving can be achieved by approximating the warp for only the vertical coordinates because of the prevalent distortion in the vertical direction over the horizontal direction. This yields an enormous time saving of about 99%. The second set of results shown compare images warped by (i) evaluating the warping function of equations 45 and 46 directly, (ii) using linear interpolation that approximates warping in both image directions, and (iii) using linear interpolation that approximates warping in only the vertical direction. Peak NCC values obtained from figures 65 (a), (b) and (c) reveal that time-saving image warping approximation methods (2) and (3) are excellent choices to use for this work.

**Table 9: Processing time saving resulting from using linear interpolation to warp most of the projected image of size 640 x 480 pixels rather than evaluating the warping function to warp the whole image**

Operation	Time to complete (s)			
	Direct evaluation of equation 45 and 46 (1)	Linear interpolation in 2 directions (2)	Linear interpolation in vertical direction (3)	% Reduction $= \frac{(1)-(3)}{(1)} \times 100$
Calculation of the warped coordinates of the projected image	20.6773	0.2504	0.1408	99.32%



**(a) Warped projected image obtained from evaluating the RBF warping function of equations 45 and 46 directly**



**(b) Warped projected image obtained from evaluating the RBF warping function of equations 45 and 46 for only the top and bottom edges of the image and applying linear interpolation to approximate the warping of the pixels in between. Peak NCC = 0.9998.**



**(c) Warped projected image obtained from evaluating the RBF warping function of equations 45 and 46 for only the top and bottom edges of the image and applying linear interpolation to approximate the warping of only vertical pixel coordinates. Peak NCC = 0.9998.**

**Figure 65: Comparison between 3 methods used to warp the 640 x 480 pixel sized projected image show that combined evaluation of the RBF warping function with linear interpolation yields very identical results.**

Table 10 is a summary of the operations with their timings. Comparing Table 10 with Table 5 shows that significant improvement has been made to operations like RBF warping, and corner and edge detection which took too long to run. The total distortion-correction processing time achieved after code profiling as seen in Table 10 is 0.3829s, yielding an operating framerate of about 2.6fps. This framerate is 60 times higher than the previous framerate of 0.047fps achieved without optimising the distortion-correction process. More significant improvement on the framerate may be achieved by a faster machine and taking advantage of parallel processing capabilities.

**Table 10: Summary table showing all image processing time savings obtained from code profiling for real-time performance**

Image	Description of group of Operations	Total Time (s)	Description of Operation	Time taken to run (s)	% Total time taken to run
<b>CAMERA-OBSERVED IMAGE</b>	Detection of 52 feature points, Homography estimation and transformation	0.0705	Detection and arrangement of 4 corner points	0.0009	0.23%
			Detection and arrangement of 48 other edge points	0.0420	10.97 %
			Estimation of projected/observed image homographies and transformation of 52 feature points from the observed image to the projected image	0.0276	7.21%
<b>PROJECTED IMAGE</b>	<i>Estimate Control points, RBF coefficients and warp RGB image</i>	0.3124	Estimation of 144 other control points from 52 to produce 196 control points	0.0008	0.21%
			Calculation of 398 RBF coefficients from 196 control points	0.0399	10.42%
			Calculation of the warped coordinates of the projected image	0.1619	42.28%
			Bilinear interpolation of warped greyscale image values	0.0917	23.95%
			Updating projected image	0.0181	4.73%
	<i>Total time</i>	0.3829			

## 5.7 Golf simulation

In simulated golf, the player strikes the golf ball with the club and the ball strikes the non-planar projection surface. It is desired that the image of the golf course on the projection surface remains geometrically undistorted typically from the view of the player. In this work the golf system is simulated by striking the projection surface with a squeezed paper projectile and correcting the resulting geometric distortion of the projected image before, during, and after the surface is hit by the projectile. The block diagram of figure 66 shows the distortion-correction process.

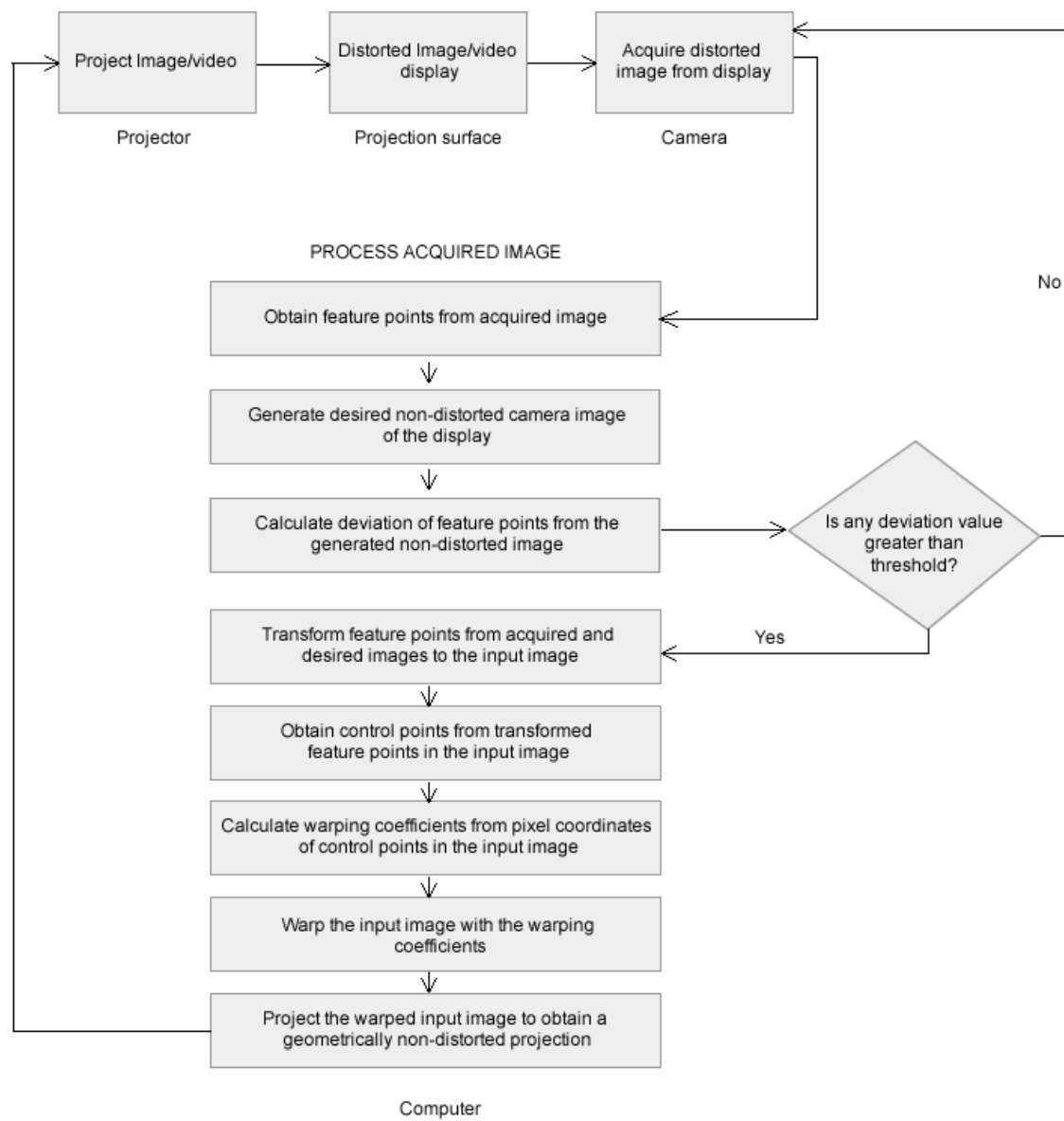


Figure 66: Geometric distortion correction process for simulated golf



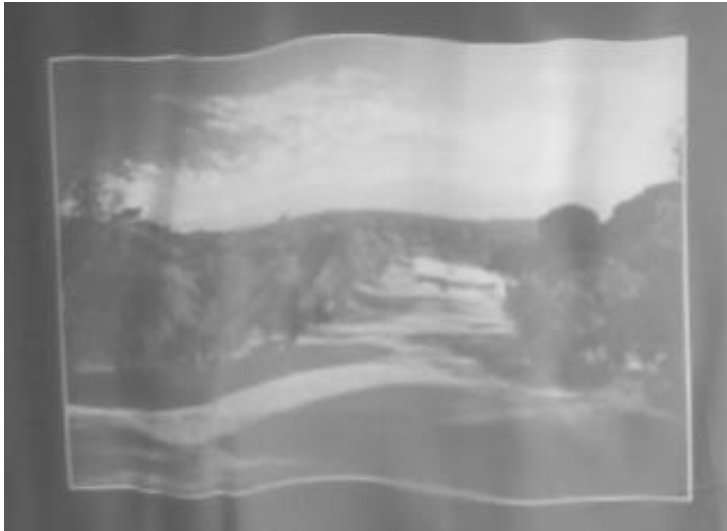
The process of geometric distortion correction outlined in figure 66 is similar to figure 5. The difference between both processes is that the system is made to continuously monitor significant distortion of the image frames of the projection obtained by the camera. This condition coincides with when the projectile strikes the projection surface. The system continues to monitor the distortion from frame to frame until the projection surface comes to rest. The distortion-correction routine is activated when the displacement of feature points from their mean position goes higher than a set threshold of 4 pixels horizontally and vertically. This threshold was determined by striking the projection surface with the projectile and recording horizontal and vertical displacements of edges of the observed projection as described in Section 5.5.1. The results of this simulation are presented in Section 5.8.

## 5.8 Results from Golf Simulation

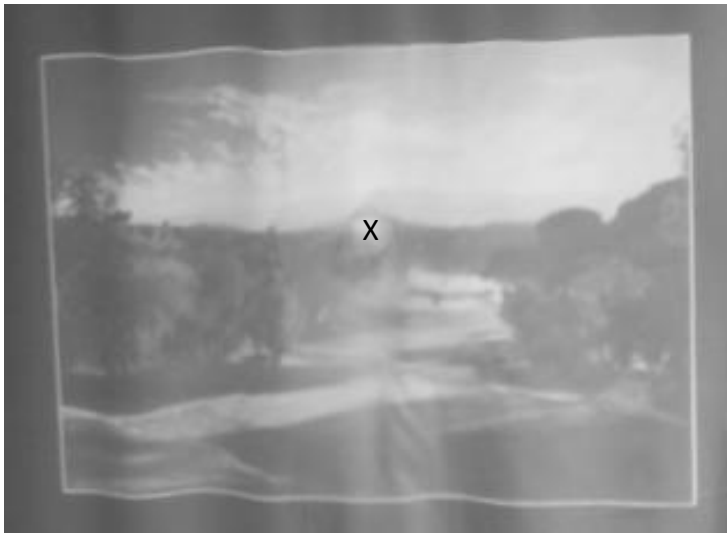
The results in Figure 67 show the projected greyscale image of the golf course, one distorted frame and three consecutive distortion-corrected frames of the projection in greyscale. The corrected images reveal that the distortion-correction system worked well with improved peak NCC values. Figure 68 shows a plot of peak NCC values for consecutive video frames, with 1 frame captured before, 3 frames captured during, and 3 others captured after the projectile hit the projection surface. The red curve with higher peak NCC values reveals successful distortion-correction while the projectile hit the surface.



(a) Projected image of golf course



**(b) Distorted image of the projection of the image of the golf course**



**(c) Distortion-corrected frame 1 of the projection of the golf course. The projectile (marked 'X') can be seen in the middle of the image.**

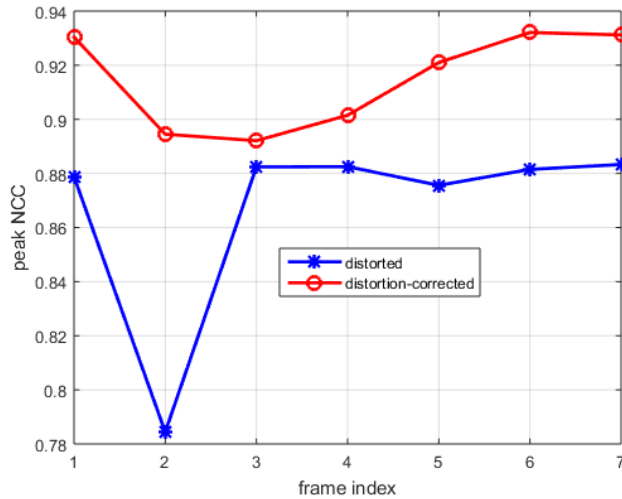


**(d) Distortion-corrected frame 2 of the projection of the golf course. The projectile can be seen in the middle of the image.**



**(e) Distortion-corrected frame 3 of the projection of the golf course. The projectile has now disappeared from view.**

**Figure 67: Projected image of a golf course and distortion-corrected frames of the projection after projectile hits the projection surface**



**Figure 68: Peak NCC values for distorted (blue) and distortion-corrected frames of the golf-simulation test**

## 5.9 Chapter Summary

The main focus of this chapter was the correction of geometric distortion for projections on dynamic/deformable non-planar displays or surfaces. The highlights drawn from the presentation include the following:

(i) The extension of the simple algebraic model of a straight line projected onto a cylinder presented in Chapter 4 to illustrate the reintroduction of observed distortion in the system distortion when the radius of the projection surface changes.

(ii) The derivation of a re-warping formula for the warped projected image to correct the observed distortion reintroduced by either changing the position of the projector and/or the observing camera or the shape of the projection surface.

The re-warping of the projected image from its original unwarped state was shown to perform better than re-warping from its already-warped state with better distortion-corrected views and consistently higher peak NCC values.

(iii) The technique of finding control points by the linear modelling of their displacements was used throughout this chapter.

(iv) Code profiling and optimisation to shorten the processing time of the distortion-correction process was carried out in this chapter.

(v) Successful optimisation strategies included introducing regions of importance (ROIs) for feature detection and estimating image corners from image edges, bilinear interpolation of a single greyscale projected image component rather than three RGB components and the calculation of warped projected image control points by linear interpolation rather than evaluating the RBF expression. Time savings from these optimisation methods achieved were 85.4%, 66.67% and 99.32% respectively. Achievable framerate was increased from 0.047fps to 2.6fps by optimisation.

(vi) All distortion-correction strategies developed for dynamic projection systems were tested on a non-planar projection surface struck by a projectile. This system was used to simulate a real projection system used in simulated golf practice. The distortion-correction process with improved peak NCC values proved to be successful in this case.

# CHAPTER SIX

## Conclusion and Recommendations

### 6.1 Conclusion

The correction of geometric distortion occurring when non-distorted images are projected on both static and dynamic/deformable non-planar surfaces has been successfully carried out in this work. By incorporating a camera as a feedback element in the projection system, images of the projection were obtained and processed to obtain geometrically non-distorted views of the projected displays.

The methods developed in this work detect feature points from both the non-distorted image being projected and the distorted image of the projection captured by the camera, derive a working homography between both images and uses this homography to shape the desired undistorted view of the projection. This process is referred to as Homography Shaping. The derived homography was used to work out the positions of the undistorted feature points on the desired camera image of the projection. It was also used to transform feature points from the camera image to control points on the projected image to be used to obtain RBF coefficients for warping the projected image to obtain an undistorted view of the projection. This method of warping the projected image to obtain an undistorted view of the projection is supported by a theoretical framework developed in this work to explain the cause and correction of the projection of a straight onto a quadric surface.

Results of tests involving the use of a calibration image consisting of a grid of white rectangles on a black background with detectable feature (corner) points show the distortion-correction method to work very well for an undulating projection surface of a freely-hanging curtain and a projection surface made of two planar surfaces intersecting at right angles. For the distortion-correction method to be successful there must be a sufficient number and distribution of feature points on the projected image and camera image of the projection. Test results also showed that the three RBFs compared performed satisfactorily in warping the projected image and correcting the distortion, but the

computing speed for the Multiquadric RBF is at least 1.14 times faster than the Inverse Multiquadric and Gaussian RBFs.

The need to have an automatic geometric distortion-correction method that adapts itself to quickly-changing projection images and projection surface shapes was emphasised in this work. A method involving the use of feature points around the periphery of the projected image and captured image of the projection, with points inside the projected image obtained by linear interpolation to meet the challenges of the dynamic projection system was developed. Tests performed with this method on both static and dynamic projection systems showed satisfactory distortion-correction results with the distortion-corrected images of the projection having consistently higher peak NCC values than the distorted images of the projection.

Drawing ROIs for feature point detection, implementing the warping function by linear interpolation rather than fully evaluating its mathematical expression, processing greyscale projected images instead of RGB images were among strategies used in this work to speed up image processing operations for geometric distortion-correction for real-time operation. By doing this, the much lower process operation time of about 21.352s was massively reduced by about 98% to 0.3592s using a Matlab GUI running on Windows 10 Operating System with an Intel Celeron N2830 2.16GHz processor. The initial frame rate of 0.04fps increased to 2.6fps. Further operating framerate increase from 2.6fps may be achieved by running the individual operations in parallel on higher specification computing platforms.

Finally, the method of distortion correction for dynamic systems was applied to a projection surface struck by the squeezed paper projectile. This test was to investigate the application of distortion-correction for projected displays of simulated golf. Very good results were also obtained from these tests with higher average peak NCC values of 0.913 obtained for captured distortion-corrected frames than average peak NCC values of 0.866 obtained for distorted frames.

## 6.2 Original Contributions

As mentioned in Sections 1.3 and 2.3.2, and demonstrated in Chapters 4 and 5, the novel successfully-tested distortion-correction techniques for non-planar projected displays developed in this work are again highlighted in this section.

(i) This work presents a geometric and algebraic explanation on how geometric distortion occurs when projecting from a plane to a non-planar surface. Here, it is shown that distortion-correction for non-planar surfaces can be achieved by the 2D homography mappings existing between the projected image and the distorted image of its projection on the projection surface captured by the camera. The warping technique of the straight line developed theoretically can be extended to projected images as well. This distortion-correction technique requiring a single uncalibrated camera and an uncalibrated projector is much simpler than systems described in [8], [27], and [31] that use calibrated projectors and cameras. This technique also eliminates the need for estimating and reconstructing the 3D projection surface as seen in [8], [31], and [30]. Eliminating these reconstruction operations and replacing them with simpler 2D homographic transformations makes it more capable of performing the distortion-correction process in real-time.

(ii) The distortion-correction technique developed in this work functions without the need for first projecting a calibration image or structured light patterns as described in [25], [28], and [29]. It was made possible by taking advantage of the nature of the non-planar projection surface that allowed for linear interpolations to be applied to displacements of already-detected distorted points from their non-distorted positions in the projected image in order to estimate the displacement of other unknown distorted points from their respective non-distorted positions. The advantage of eliminating the projection of a calibration image or structured light patterns is that in situations where the shape of the projection surface changes or the pose of the projector or/and the observing camera change, the projection does not have to be interrupted to allow for the calibration image to be re-projected onto the surface in order to maintain a distortion-corrected view of the projection.



(iii) The technique developed in Section 5.3 for correcting the distortion of dynamic projection systems comprising systems where the positions of the projector and/or the camera change and/or the shape of the projection surface changes with time is novel. A few authors ([8] and [33] for example) however mentioned the capabilities of their techniques to handle such situations without presenting any supporting results.

(iv) A much simpler RBF interpolation approach to warping as opposed to other processing-demanding warping methods like Bezier patches used in other systems is used in this work. The optimisation of the RBF image warping algorithm for a 640 x 480 pixel-sized image yielded a time saving of 99.32% (from 20.99s to 0.1619s), enhancing the potentials for the approach to be applicable to real-time operation.

## **6.3 Practical Considerations**

### **6.3.1 Practical Applications**

Practical applications of the distortion-correction system for non-planar projected displays include possible presentations in places where there is no screen and the simulated golf practice system simulated in Section 5.7. The distortion-correction system developed in this work usually applies to a single or a few viewers who share the same view of the projection surface as that of the observing camera. For viewers to enjoy a distortion-free view in a place where a non-planar projection surface is used instead of a standard planar surface, they should have similar field of view as the camera used to observe the surface. The algebraic simulation of a non-planar projection system carried out in Section 4.2 can also be used to estimate possible distortion-free viewing positions for a particular pose of the projector, camera and non-planar projection surface.

In simulated golf practice the user plays the golf ball against a deformable surface that dampens the motion of the golf ball and prevents injury to the player as the ball rebounds off the surface. The trajectory of the golf ball and some text telling its measured speed may also be displayed on the screen. The player's experience can be improved by removing the geometric distortion from the entire projection as the ball strikes the projection surface.

Incorporating the distortion-correction system with a head-worn device that tracks the position of the player's viewpoint could also improve the user's experience he/she will likely move his/her head during play.

### **6.3.2 Image Processing Considerations**

In all tests the homography between the camera image of the projection and the projected image was obtained by 4 corner point correspondences between both images. More undistorted points may have been used to estimate this homography with a refinement to the estimated homography done by suitable least-square optimisation techniques. Such techniques were not used in this work because the estimated homography proved to be good enough for the purpose and because of the increased enormous processing requirements required to implement them.

Additional control points inside the projected image had to be generated for effective pre-warping of the projected image and geometric distortion correction. These additional control points were obtained from already-detected control points located at the top and bottom edges of the projected image by linear interpolation (Sections 4.5.3 and 4.6.3. Other methods like inserting invisible patterns (watermarks) in the projected image may be used to create such additional control points. To do this a very robust method for correctly extracting the watermark from the projection must be developed to overcome the enormous radiometric modulation and geometric deformation that the projected image would go through right from the projector through the projection surface to the camera. Another method which may be used to generate additional control points is by colour segmentation and matching between the projected image and the image of the projection captured by the camera. This kind of segmentation and matching using the Hue-Saturation-Value (HSV) colour space has been reported in [91] for example.

The surface used in this work was such that images projected on it suffered geometric distortion predominantly in the vertical direction. For more complex projection surfaces with significant distortion in both directions, distortion correction may be done in one

direction first, then the other. The Linear Interpolation method developed in Section 5.5.3 used to approximate the RBF warping of the projected image may only be used in this application because of the nature of the projection surface, but it may also be worthwhile to test it in other applications where speed of operation will be of essence.

The treatment given to lens distortion would include the lens of the projector and that of the camera. The ultimate aim in all treatments given to lens distortion is to restore curved (distorted) image lines caused by the physical properties of the lens to straight lines. In this work no explicit treatment is given to lens distortion because it is assumed that the components of lens distortion add up to and are indistinguishable from the distortion caused by the shape of the projection surface, and the ultimate aim in this work is to restore distorted image lines to straight lines. With this assumption however, the lens distortion caused by the camera still causes the distortion of distortion-corrected lines in camera images of the projection and accounts for much of the unresolved distortion (relatively large average deviation values) in the results presented in chapters 4 and 5. The presence of this so-called unresolved distortion of lines in camera images of the projection is however inconsequential because the distortion-correction system is designed for human observers looking at the projection surface and the human psychovisual system corrects such geometric distortions.

#### **6.4 Recommendations**

The following recommendations given are ways of improving the technique developed in this work to correct the observed geometric distortion of non-distorted images and video projected onto static and deformable displays or surfaces. They include distortion-correction topics not addressed in this work.

(i) More complex surfaces with predominant distortion in both image dimensions: It is recommended that geometric distortions of projections on such surfaces be corrected with one dimension first, and then the other. Difficulties may arise from the area where the pre-warped projected image is being projected to as this could as well cause its own distortion.

In such a case there should be recursive correction whereby the observed projection is always monitored for distortion followed by correction by warping the projected image.

(ii) Strategies to improve viewer satisfaction: In the case of a single viewing experience, multiple feedback cameras may be used with a single projector and each camera would be used to observe and correct its respective distortion of the projection. If possible, each pre-warped image should be beamed onto the projection surface at a refresh rate that will not be noticed by the audience.

(iii) Improvement of the quality of the projection: It is observed that the quality of images of the distortion-corrected projection presented in this work are deteriorated by shadows caused by the shape of the projection surface. Some photometric enhancements like shadow suppression and contrast, colour and brightness enhancements may be used to enhance the quality of the distortion-correction system.

# APPENDIX

## A1 Detection and Arranging Detected Corner Points of the Distorted Projected Calibration Image

The success of the use of a calibration image in the distortion-correction process greatly depends on the accurate detection and arrangement of the feature (corner) points of the captured image of its projection. Unfortunately the output of the inbuilt Harris corner detector function in Matlab produces the detected corners in an unordered form. A technique of correctly arranging the detected corner points was carried out in this work and is described as follows:

### Problem

An unordered list of corner point  $(u, v)$  pixel coordinates. The task is to arrange them according to the way they appear in the projected image.

### Solution

(i) First arrange them in increasing distance from the origin by calculating the distance between each point and the image origin  $(0,0)$  by calculating the square root of the sum of the square of the coordinates of the respective points. This step simplifies the whole process by ensuring that the first point on the list is the corner point at the top left corner of the image.

$$\text{sort}\sqrt{(u^2 + v^2)} \quad (64)$$

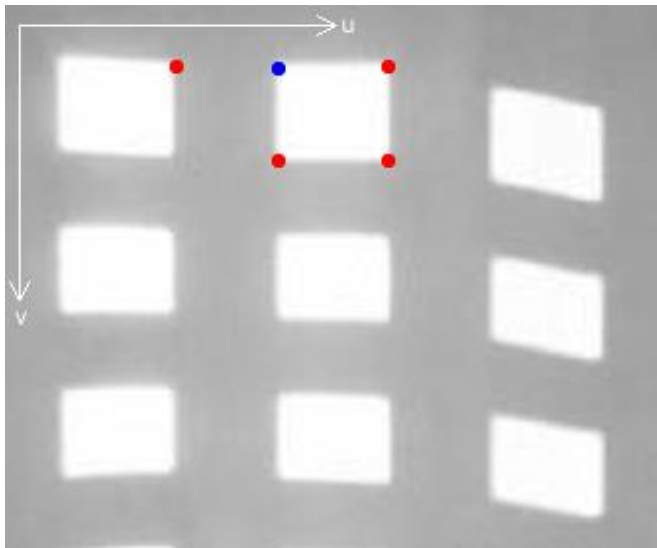
Since the actual distance of each point from the origin is not required, computation time may be saved by simply ranking the points by the sum of their respective coordinates.

Next, calculate the mutual distances between the respective points as

$$\sqrt{(u_i - u_j)^2 + (v_i - v_j)^2} \quad (65)$$

The aim of this step and others following is to have all points arranged according to the horizontal (or vertical) lines they belong to by finding their closest horizontal and vertical neighbours. An alternative to equation 65 is to calculate the sum of the unsigned differences between the coordinates of the respective points.

(ii) Select a point and find its horizontally, vertically diagonally-aligned neighbours by sorting results obtained from equation 65 for the chosen point. Three or four points connected to the chosen point are obtained from this step. A corner point and its four closest neighbours are shown in figure 69.



**Figure 69: A chosen corner point (blue) and its four closest neighbours**

(iii) If horizontally-aligned points are being sought for example, sort the closest neighbours obtained in step (ii) according to their respective vertical distances to the point of interest. The desired horizontally-aligned neighbour will be the point with the smallest vertical distance to the point of interest. If a vertically-aligned neighbour is required, sought-after vertical neighbour will be the point with the smallest horizontal distance to the point of interest.

(iv) As one of the selected neighbours has been found as a collinear point to the reference point, exclude the reference point from the list of points to be searched, and make the collinear point just found the new reference. Return to step (ii) and continue until the set number of points per line is reached.

(v) Sort the points on the line according to value of their horizontal pixel coordinates to arrange them from left to right.

Next, calculate the average vertical coordinate of all the vertical coordinates of the points on the line. This is used as a summary statistic representing the position of the line in a stack of horizontal lines.

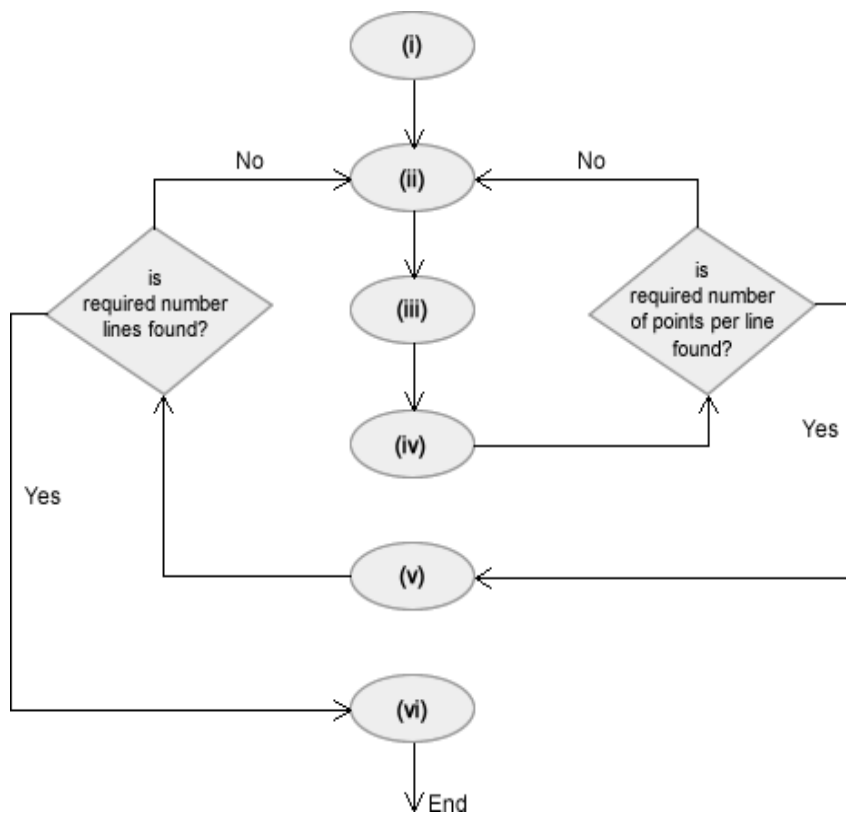
Return to step (ii) and continue the same process for other horizontal lines, finding neighbouring points and arranging them according to how they are located along the line.

(vi) When the required number of horizontal lines is reached, use the summary statistic calculated in step (v) to arrange the lines from top to bottom in increasing order of the statistic.

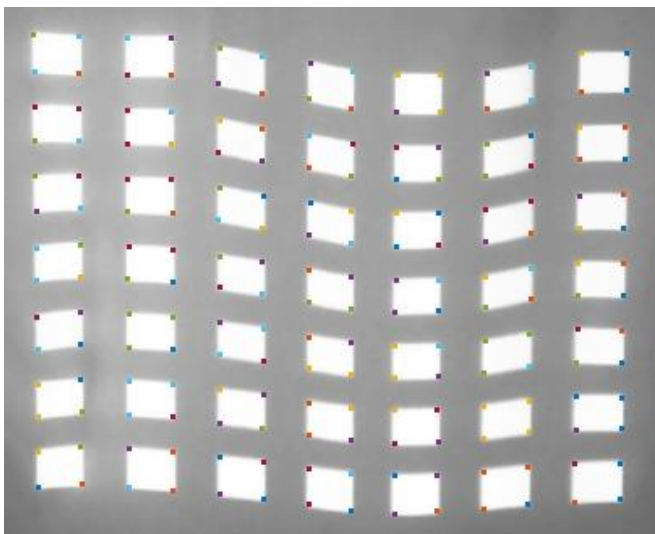
A flow diagram representing the process described is shown in figure 70.

Note that this method produces the correct results only in the absence of false or noisy corner points and/or missing corner points.

Figure 71 shows the captured view of the calibration image with misarranged corners in (a) and arranged corners in (b).

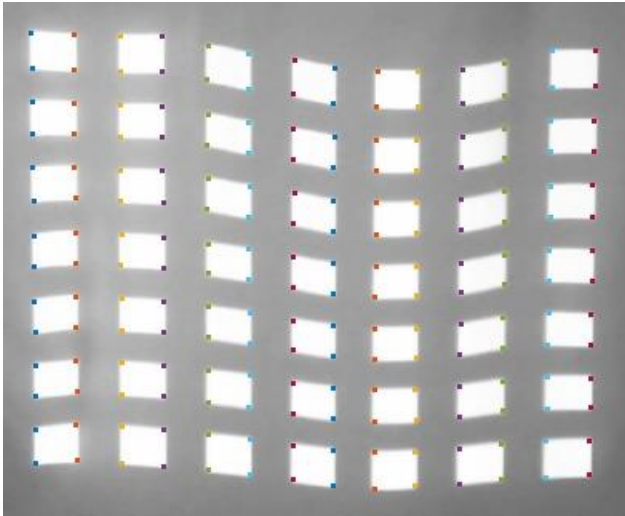


**Figure 70: Flow diagram of steps used in producing an ordered (arranged) set of corner points from an unordered (misarranged) set of points**



**(a) Misarranged corners**





**(b) Correctly arranged corners**

**Figure 71: Camera view of the calibration image showing misarranged and correctly-arranged corners**

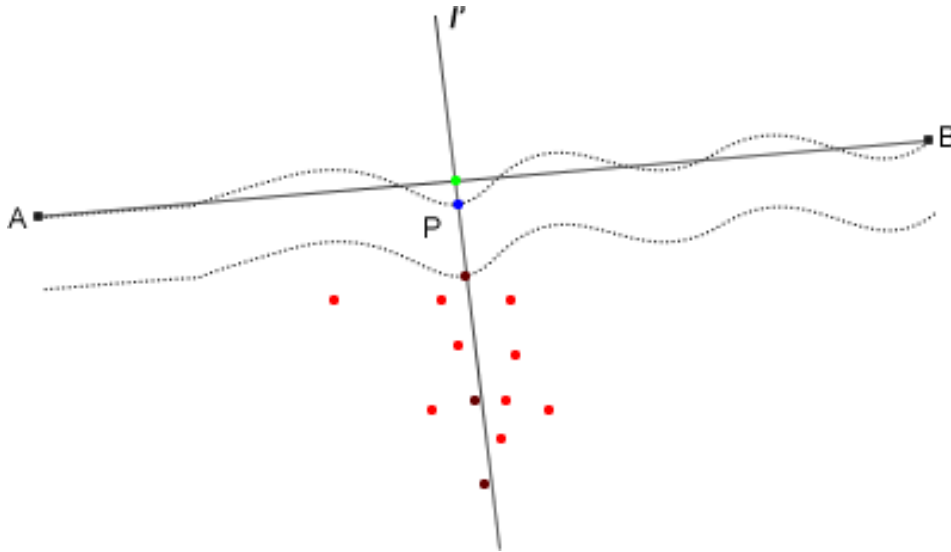
## **A2 Obtaining Feature Points along a Distorted Edge**

In Section 4.5 and subsequently, the homography between the projected image and camera image of the projection was used to obtain non-distorted feature points on the latter image by transforming the points from an ordered phantom grid of equal size with the projected image. These non-distorted points are then used to obtain their respective distorted points along the top and bottom edges of the image of the projection (as described in steps (iv) and (v) of Section 4.5.1).



**Figure 72: Edge image of the projection showing red non-distorted points of the blue top and bottom edges**

The edge image shown figure 72 is the input image to the algorithm, consisting of the edges in white and the non-distorted sampled points in red. Only a single point out of thousands of edge points must be selected for each non-distorted point. A typical (top) section of the periphery of figure 72 is redrawn in figure 73 for clarity. It illustrates the respective lines and points described in the following guide to selecting the required feature.



**Figure 73: A typical top peripheral section of the camera image of the projection showing the distorted edge with the blue distorted feature point, the desired non-distorted edge (line AB) with the green non-distorted feature point, the perpendicular line  $l'$ , and other detected feature points in the image.**

The guide to selecting the appropriate points is given as follows:

(i) Obtain the equation of the line linking the two extreme corner points forming the ends of the non-distorted top and bottom edges of the image with equation 25. (Line  $AB$  of figure 73).

(ii) Obtain the equation of the perpendicular line  $l'$  to line  $AB$  passing through the point  $P$  (equation 29).

(iii) Test each feature point to see which one falls on the line by inputting its horizontal and vertical coordinates into the equation of the perpendicular line. The few eligible points (blue and brown points in figure 73) are those whose coordinates satisfy the equation of the line, or are close enough to a set threshold.

From equation 25,

$$\left| v - \frac{v_1 - v_2}{u_1 - u_2} u - \frac{u_1 v_2 - u_2 v_1}{u_1 - u_2} \right| \leq \text{threshold} \quad (66)$$

(iv) From the selected points in step (iii) calculate the distance of each point to the line  $AB$  (with equation 27) and select the 2 points with the shortest distances from the line. These will be the blue point and the topmost brown point in figure 73. Selecting 2 points in this work arises from the fact that depending on the extent of distortion, a point from any of the enhanced edges (edges forming an inner and outer boundary in figure 73) would be detected.

(v) The required point on the top edge (blue point in figure 73) would be one of the points from step (iv) with the lower vertical coordinate while the required point on the bottom edge would be the point from step (iv) with the higher vertical coordinate.

### **A3 Graphical User Interface (GUI)**

The GUI created in Matlab and used for all the tests in this work is shown in figure 74.

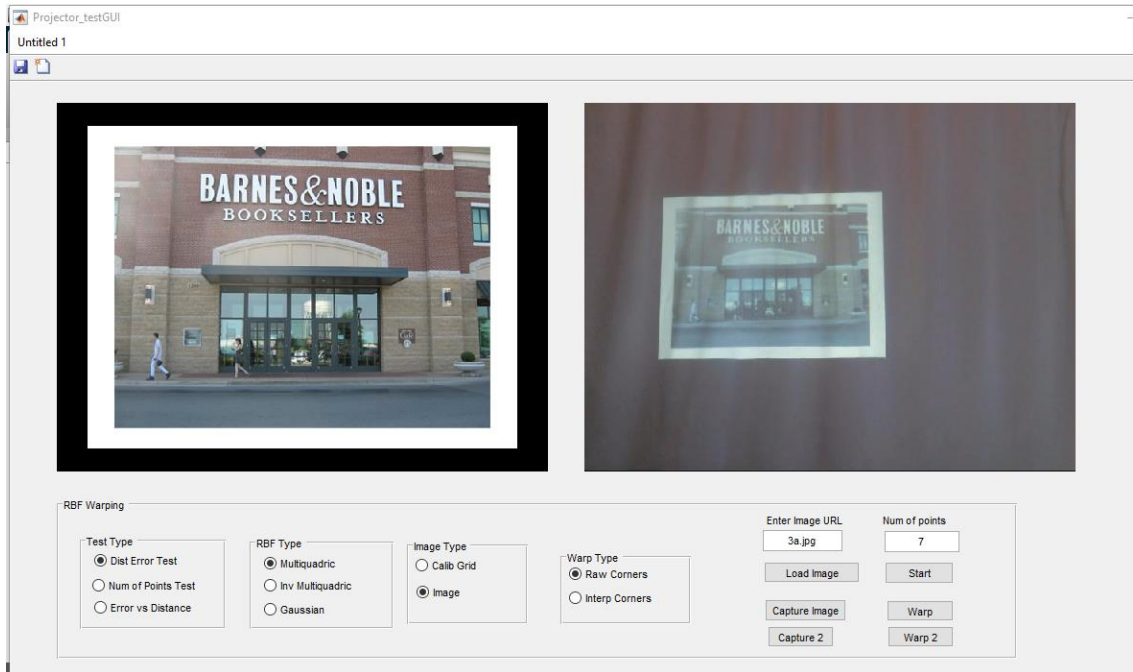


Figure 74: GUI created in Matlab

## REFERENCES

- [1] M. B. Stegmann, "Image Warping," 21 October 2001. [Online]. Available: <https://cs.brown.edu/courses/csci1950-g/asgn/proj6/resources/ImageWarping.pdf>. [Accessed 25 November 2015].
- [2] H. Fuchs, M. A. Livingston, R. Raskar, D. Colucci, K. Keller, A. State, J. R. Crawford, P. Rademacher, S. H. Drake and A. A. Meyer, "Augmented Reality Visualization for Laparoscopic Surgery," in *Medical Image Computing and Computer-Assisted Intervention*, Berlin Heidelberg, Springer, 1998, pp. 934-943.
- [3] R. Raskar, G. Welch and C. Wei-Chao, "Table-top Spatially-Augmented Reality: Bringing Physical Models to Life with Projected Imagery," in *2nd IEEE and ACM International Workshop on Augmented Reality*, San Francisco, CA, 1999.
- [4] M. D. Grossberg, H. Peri, S. K. Nayar and P. N. Belhumeur, "Making one Object Look like Another: Controlling Appearance Using a Projector-Camera System," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2004.
- [5] B. Coxworth, "Transparent Back seat Rearview System," Gizmag, 8 November 2012. [Online]. Available: <http://www.gizmag.com/transparent-back-seat-rearview-system/24943/>. [Accessed 19 December 2015].
- [6] C. J. Bair, R. J. Bair and L. R. Corn, "Golf Simulator". United States Patent 5,846,139, 8 December 1998.
- [7] "The Ultimate Golf Simulators Buyer Guide," 9 January 2014. [Online]. Available: <http://www.golf-simulators-buying-guide.com/>. [Accessed 19 December 2015].
- [8] K. Ito, T. Takahashi and T. Aoki, "A Geometric Correction Method Using Stereo Vision for Projected Images," in *First Asian Conference on Pattern Recognition*, 2011.

- [9] L. Zhang, A. M. Yip and C. L. Tan, "Photometric and Geometric Restoration of Document Images Using Inpainting and Shape from Shading," in *22nd Conference on Artificial Intelligence*, 2007.
- [10] H. Cao, X. Ding and C. Liu, "A Cylindrical Surface Model to Rectify the Bound Document Image," in *Proceedings of the Ninth IEEE International Conference on Computer Vision*, Nice, France, 2003.
- [11] M. S. Brown and Y. Tsoi, "Geometric and Shading Correction for Images of Printed Materials Using Boundary," *IEEE Transactions on Image Processing*, vol. 15, no. 6, pp. 1544-1554, 2006.
- [12] M. S. Brown and W. B. Seales, "Image Restoration of Arbitrarily Warped Documents," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 10, pp. 1295-1306, 2004.
- [13] K. Seemakurthy and A. N. Rajagopalan, "Deskewing of Underwater Images," *IEEE Transactions on Image Processing*, vol. 24, no. 3, pp. 1046-1059, 2015.
- [14] T. Naruse, T. Kaneko, A. Yamashita and H. Asama, "3D Measurement of Objects in Water Using Fish-eye Stereo Camera," in *19th IEEE International Conference on Image Processing*, Orlando, FL, 2012.
- [15] H. Murase, "Surface Shape Reconstruction of an Undulating Transparent Object," in *Proceedings of the Third International Conference on Computer Vision*, Osaka, 1990.
- [16] R. Swaminathan and S. K. Nayar, "Nonmetric Calibration of Wide-Angle Lenses and Polycameras," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 10, pp. 1172-1178, 2000.

- [17] R. Y. Tsai, "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. 3, no. 4, pp. 323-344, 1987.
- [18] L. Alvarez, L. Gomez and J. R. Sendra, "An Algebraic Approach to Lens Distortion by Line Rectification," *Journal of Mathematical Imaging and Vision*, vol. 35, no. 1, pp. 36-50, 2009.
- [19] R. Sukthankar, G. Sukthankar and C. Tat-Jen, "Dynamic Shadow Elimination for Multi-projector Displays," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2001.
- [20] J. Liao and L. Cai, "A Calibration Method for Uncoupling Projector and Camera of a Structured Light System," in *IEEE/ASME International Conference on Advanced Intelligence Mechatronics*, 2008.
- [21] D. Moreno and G. Taubin, "Simple, Accurate, and Robust Projector-Camera Calibration," in *2nd International Conference on 3D Image Modeling, Processing, Visualisation and Transmission*, 2012.
- [22] M.-H. Lee, H. Park and J. Park, "Fast Radiometric Compensation Accomplished by Eliminating Color Mixing between Projector and Camera," *IEEE Transactions on Consumer Electronics*, vol. 54, no. 3, pp. 987-991, 2008.
- [23] R. Raskar and P. Beardsley, "A Self-correcting Projector," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2001.
- [24] B. Li and I. Sezan, "Automatic Keystone Correction for Smart Projectors with Embedded Camera," in *International Conference on Image Processing*, 2004.

- [25] J.-P. Tardif, S. Roy and M. Trudeau, "Multi-Projectors for Arbitrary Surfaces Without Explicit Calibration nor Reconstruction," in *Fourth International Conference on 3D Digital Imaging and Modeling*, 2003.
- [26] M. Harville, B. Culbertson, I. Sobel, D. Gelb and D. T. Fitzhugh, "Practical Methods for Geometric and Photometric Correction of Tiled Projector Displays on Curved Surfaces," in *Conference on Computer Vision and Pattern Recognition Workshop*, 2006.
- [27] H. Park, M.-H. Lee, B.-K. Seo, J.-I. Park, M.-S. Jeong, T.-S. Park, Y. Lee and S.-R. Kim, "Simultaneous Geometric and Radiometric Adaptation to Dynamic Surfaces with a Mobile Projector-Camera System," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 1, pp. 110-115, 2008.
- [28] T.-J. Yang, Y.-M. Tsai and L.-G. Chen, "Smart Display: A Mobile Self-Adaptive Projector-Camera System," in *IEEE International Conference on Multimedia and Expo*, 2011.
- [29] C. Xie, Q. Wang and W. Cheng, "Simple Auto-Geometric Correction for Non-Planar Projection," in *International Conference on Automatic Control and Artificial Intelligence*, 2012.
- [30] B. Sajadi and A. Majumder, "Markerless View\_Independent Registration of Multiple Distorted Projectors on Extruded Surfaces Using an Uncalibrated Camera," *IEEE Transactions on Visualisation and Computer Graphics*, vol. 15, no. 6, pp. 1307-1316, 2009.
- [31] T. Yamanaka, F. Sakue and J. Sato, "Adaptive Image Projection onto Non-Planar Screen Using Projector-Camera Systems," in *International Conference on Pattern Recognition*, 2010.
- [32] K. Nishie and J. Sato, "3D Reconstruction from Uncalibrated Cameras and Uncalibrated Projectors from Shadows," in *International Conference on Pattern Recognition*, 2006.



- [33] F. Sun and W. Meng, "An Interactive Warping Method for Multi-Channel VR Projection Display Systems with Quadric Surface Screens," in *International Conference on Virtual Reality and Visualisation*, 2013.
- [34] N. Arad, N. Dyn, D. Reifeld and Y. Yeshurun, "Image Warping by Radial Basis Functions: Application to Facial Expressions," *Graphical Models and Image Processing*, vol. 56, no. 2, pp. 161-172, March 1994.
- [35] C. A. Glasbey and K. V. Mardia, "A Review of Image Warping Methods," *Journal of Applied Statistics*, vol. 25, pp. 155-171, 1998.
- [36] A. Goshtasby, "Piecewise Linear Mapping Functions for Image Registration," *Pattern Recognition*, vol. 19, no. 5, pp. 459-466, 1986.
- [37] J. A. Little, D. L. Hill and D. J. Hawkes, "Deformations Incorporating Rigid Structures," in *IEEE Proceedings of MMBIA*, 1996.
- [38] J. Montagnat, H. Delingette and N. Ayache, "A Review of Deformable Surfaces: Topology, Geometry and Deformation," *Image and Vision Computing*, vol. 19, no. 14, pp. 1023-1040, 2001.
- [39] S. F. Gibson and B. Mirtich, "A Survey of Deformable Modeling in *Computer Graphics*," Cambridge, Massachusetts, 1997.
- [40] E. S. Bhasker and A. Majumder, "Geometric Modeling and Calibration of Planar Multi-Projector Displays Using Rational Bezier Patches," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2007.
- [41] Z. Qiu, H. Tang and D. Tian, "Non-rigid medical image registration based on the thin-plate spline algorithm," in *WRI World Conference on Computer Science and Information Engineering*, 2009.

- [42] W. Qin, Y. Hu, Y. Sun and B. Yin, "An Automatic Multi-sample 3D Face Registration Method Based On Thin Plate Spline and Deformable Model," in *IEEE International Conference on Multimedia and Expo Workshops*, 2012.
- [43] O. Cakmakci, G. E. Fasshauer, H. Foroosh, K. P. Thompson and J. P. Rolland, "Meshfree Approximation Methods for Free-form Surface Representation in Optical Design with Applications to Head-worn Displays," in *SPIE Proceedings*, 2008.
- [44] A. Bauer, S. Vo, K. Parkins, F. Rodriguez, O. Cakmakci and J. Rolland, "Computational Optical Distortion Correction Using a Radial Basis Function-based Mapping Method," *Optics Express*, vol. 20, no. 14, pp. 14906-14920, 2012.
- [45] J. C. Carr, W. R. Fright and R. K. Beatson, "Surface Interpolation with Radial Basis Functions for Medical Imaging," in *IEEE Transactions on Medical Imaging*, 1997.
- [46] V. Skala, "Fast Reconstruction of Corrupted Images and Videos by Radial Basis Functions," in *International Conference on Control, Automation and Information Sciences (ICCAIS)*, 2013.
- [47] S. Jakobsson and O. Amoignon, "Mesh Deformation Using Radial Basis Functions for Gradient-based Aerodynamic Shape Optimisation," Stockholm, 2005.
- [48] J.-Y. Noh, D. Fidaleo and U. Neumann, "Animated deformations with radial basis functions," in *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, New York, 2000.
- [49] B. Zitova and J. Flusser, "Image Registration Methods: A Survey," *Image and Vision Computing*, vol. 21, no. 11, pp. 977-1000, October 2003.
- [50] K. Briechle and U. D. Hanebeck, "Template Matching Using Fast Normalised Cross-correlation," in *Proc. SPIE 4387, Optical Pattern Recognition XII*, 95, 2001.

- [51] R. B. Fisher and P. Oliver, "Multivariate Cross-correlation and Image Matching," in *British Machine Vision Conference*, 1995.
- [52] H. Hanaizumi and S. Fujimura, "An Automated Method for Registration of Satellite Remote Sensing Images," in *International Geoscience and Remote Sensing Symposium*, Tokyo, Japan, 1993.
- [53] W. K. Pratt, *Digital Image Processing*, Los Altos, California: John Wiley and Sons inc, 2007.
- [54] Mathworks Inc, *Matlab Image Processing Toolbox Userguide*, Natick, MA: The Mathworks Inc, 1993-2013.
- [55] R. Maini and H. Aggarwal, "Study and Comparison of Various Image Edge Detection Techniques," *International Journal of Image Processing*, vol. 3, no. 1, pp. 1-12, 2009.
- [56] S. Bhardwaj and A. Mittal, "A Survey on Various Edge Detection Techniques," in *2nd International Conference on Computer, Communication, Control and Information Technology*, 2012.
- [57] R. Fisher, S. Perkins, A. Walker and E. Wolfart, "Image Processing Learning Resources," *Hypermedia Image Processing Reference*, 2003. [Online]. Available: <http://homepages.inf.ed.ac.uk/rbf/HIPR2/log.htm>. [Accessed 30 August 2015].
- [58] J. Canny, "A Computational Approach to Edge Detection," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679-698, 1986.
- [59] Rashmi, M. Kumar and R. Saxena, "Algorithm and Technique on Various Edge Detection: A Survey," *Signal and Image Processing: An International Journal*, vol. 4, no. 3, pp. 65-75, 2013.

- [60] D. Parks and J. Gravel, "Corner Detection," [Online]. Available: <http://nartoolkit.googlecode.com/svn/trunk/Documentation/Corners/CornerDetection.pdf>. [Accessed 1 September 2015].
- [61] Z. Zhang, "A flexible new technique for camera calibration," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1330-1334, 2000.
- [62] R. Bhalerao and S. S. Gedam, "Disparity Computation of Corner Points by Enhancing Feature Extractors for Sparse Narrow-Baseline Stereo Matching," in *IEEE International Geoscience and Remote Sensing Symposium*, 2011.
- [63] M. Liu, W. Chengdong and Y. Zhang, "Motion Vehicle Tracking Based on Multi-Resolution Optical Flow and Multi-Scale Harris Corner Detection," in *IEEE International Conference on Robotics and Biomimetics*, Sanya, 2007.
- [64] Y. Pei, H. Wu, J. Yu and G. Cai, "Effective Image Registration Based on Improved Harris Corner Detection," in *International Conference on Information Networking and Automation*, Kunming, 2010.
- [65] B. S. Kim, S. H. Lee and N. I. Cho, "Real-time Panorama Canvas of Natural Images," *IEEE Transactions on Consumer Electronics*, vol. 57, no. 4, pp. 1961-1968, 2011.
- [66] C. Gao, K. Panetta and S. Agaian, "Robust Template based Corner Detection Algorithms for Robotic Vision," in *IEEE International Conference on Technologies for Practical Robot Applications*, Woburn, MA, 2015.
- [67] F. Mokhtarian and F. Mohanna, "Performance Evaluation of Corner Detectors using Consistency and Accuracy Measures," *Computer Vision and Image Understanding*, vol. 102, pp. 81-94, 2006.

- [68] H. Asada and M. Brady, "The Curvature Primal Sketch," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 1, pp. 2-14, 1986.
- [69] H. P. Moravec, "Towards Automatic Visual Obstacle Avoidance," in *International Joint Conference on Artificial Intelligence*, 1977.
- [70] J. Liu, A. Jakas, A. Al-Obaidi and Y. Liu, "A Comparative Study on Different Corner Detection Methods," in *IEEE International Symposium on Computational Intelligence in Robotics and Automation*, Daejeon, 2009.
- [71] C. Harris and M. Stephens, "A Combined Corner and Edge Detector," in *Alvey Vision Conference*, 1988.
- [72] S. M. Smith and J. M. Brady, "SUSAN - A New Approach to Low Level Image Processing," *International Journal of Computer Vision*, vol. 23, no. 1, pp. 45-78, 1997.
- [73] E. Rosten and T. Drummond, "Machine Learning for High-speed Corner Detection," in *European Conference on Computer Vision*, 2006.
- [74] J. Shi and C. Tomasi, "Good Features to Track," in *IEEE Conference on Computer Vision and Pattern Recognition*, Seattle, 1994.
- [75] Wikipedia contributors, "Corner Detection," Wikipedia, The Free Encyclopedia, 29 August 2015. [Online]. Available:  
[https://en.wikipedia.org/w/index.php?title=Corner\\_detection&oldid=678472890](https://en.wikipedia.org/w/index.php?title=Corner_detection&oldid=678472890).  
[Accessed 2 September 2015].
- [76] Wikipedia contributors, "Blob Detection," Wikipedia, the Free Encyclopedia, 10 May 2015. [Online]. Available:  
[https://en.wikipedia.org/w/index.php?title=Blob\\_detection&oldid=661683516](https://en.wikipedia.org/w/index.php?title=Blob_detection&oldid=661683516). [Accessed 2 September 2015].

- [77] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91-110, 2004.
- [78] M. Sonka, V. Hlavac and R. Boyle, *Image Processing, Analysis, and Machine Vision*, 3rd ed., Stamford, CT: Cengage Learning, 2008.
- [79] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed., New York: Cambridge University Press, 2004.
- [80] M. Kimura, M. Mochimaru and T. Kanade, "Projector Calibration using Arbitrary Planes and Calibrated Camera," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2007.
- [81] J. Draréni, S. Roy and P. Sturm, "Methods for Geometrical Video Projector Calibration," *Machine Vision and Applications*, vol. 23, pp. 79-89, 2012.
- [82] T. M. C. Shen, "Digital Projector Calibration for 3D Active Vision Systems," *Journal of Manufacturing Science and Engineering*, vol. 124, no. 1, pp. 126-134, April 2000.
- [83] J. Tao, "Slide Projection Calibration Based on Calibration of Digital Camera," in *Proc. SPIE 6788, MIPPR 2007: Pattern Recognition and Computer Vision*, 2007.
- [84] R. Sukthankar, R. G. Stockton and M. D. Mullin, "Smarter Presentations: Exploiting Homography in Camera-Projector Systems," in *Proceedings of International Conference on Computer Vision*, Vancouver, BC, 2001.
- [85] Wikipedia contributors, "Radial Basis Function," Wikipedia, The Free Encyclopedia, 14 May 2015. [Online]. Available: [https://en.wikipedia.org/w/index.php?title=Radial\\_basis\\_function&oldid=662370301](https://en.wikipedia.org/w/index.php?title=Radial_basis_function&oldid=662370301). [Accessed 28 August 2015].

- [86] A. de Boer, M. van der Schoot and H. Bijl, "New Methods of Mesh Moving Based on Radial Basis Function Interpolation," in *European Conference on Computational Fluid Dynamics*, Delft, The Netherlands, 2006.
- [87] Y. R. Rao, N. Prathapani and E. Nagabhooshanam, "Application of Normalised Cross Correlation to Image Registration," *International Journal of Research in Engineering and Technology*, vol. 3, no. 5, pp. 12-16, 2014.
- [88] K. Briechle and U. D. Hanebeck, "Template Matching Using Fast Normalised Cross Correlation," *Proceedings of SPIE-The International Society of Optical Engineering*, vol. 3, 2001.
- [89] H. Foroosh and J. B. Zerubia, "Extension of Phase Correlation to Subpixel Registration," *IEEE Transactions on Image Processing*, vol. 11, no. 3, pp. 188-200, 2002.
- [90] H. Sterk, "Geometry in Architecture and Building," 2008. [Online]. Available: <http://www.win.tue.nl/~sterk/Bouwkunde/hoofdstuk2.pdf>. [Accessed 5 September 2015].
- [91] S. Vitabile, G. Pollaccia, G. Pilato and F. Sorbello, "Road Signs Recognition Using a Dynamic Pixel Aggregation Technique in the HSV Color Space," in *11th International Conference on Image Analysis and Processing*, 2001.
- [92] J. M. Fitzpatrick, D. L. Hill and C. R. Maurer, "Image Registration," in *Handbook of Medical Imaging 2*, Chicago, 2000, pp. 447-513.
- [93] M. Kimura, M. Mochimaru and T. Kanade, "Projector Calibration using Arbitrary Planes and Calibrated Camera," in *IEEE Conference on Computer Vision and Pattern Recognition*, Minneapolis, 2007.

- [94] T. Okatani and K. Deguchi, "Autocalibration of a Projector-Camera System," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 12, pp. 1845-1855, 2005.
- [95] D. Crasto, A. Kale and C. Jaynes, "The Smart Bookshelf: A Study of Camera Projector Scene Augmentation of an Everyday Environment," in *7th IEEE Workshops on Application of Computer Vision*, 2005.
- [96] H. Park, M.-H. Lee, S.-J. Kim and J.-I. Park, "Surface-Independent Direct-Projected Augmented Reality," in *Computer Vision - ACCV 2006*, Springer Berlin Heidelberg, 2006, pp. 892-901.
- [97] J. Wu, Z. Cui, V. Sheng, P. Zhao, D. Su and S. Gong, "A Comparative Study of SIFT and its Variants," *Measurement Science Review*, vol. 13, no. 3, pp. 122-131, 2013.