Seyed Jafar Hosseini Shamoushaki

DEFORMABLE SURFACE RECONSTRUCTION THROUGH OPTIMIZATION-BASED TECHNIQUES

PhD thesis in Electrical and Computer Engineering, the specialization branch of Automation and Robotics, supervised by Professor Doctor Helder de Jesus Araujo and submitted to the Faculty of Science and Technology of the University of Coimbra

September, 2016

• U C •

UNIVERSIDADE DE COIMBRA

University of Coimbra

Faculty of Science and Technology

Department of Electrical and Computer Engineering

Deformable Surface Reconstruction Through Optimization-Based Techniques

PhD thesis in Electrical and Computer Engineering,

the specialization branch of Automation and Robotics,

supervised by Professor Doctor Helder de Jesus Araujo and

submitted to the Faculty of Science and Technology of the University of Coimbra.

PhD Thesis

Seyed Jafar Hosseini Shamoushaki

Coimbra, September 2016

University of Coimbra

Faculty of Science and Technology

Department of Electrical and Computer Engineering

Deformable Surface Reconstruction Through Optimization-Based Techniques

Thesis submitted:

to the Electrical and Computer Engineering Department of the Faculty of Science and Technology of the University of Coimbra in partial fulfillment of the requirements for the Degree of Doctor of Philosophy.

Seyed Jafar Hosseini Shamoushaki

Coimbra, September 2016

This thesis has been supervised by

Professor Doctor Helder de Jesus Araujo

Professor of the

Faculty of Science and Technology, University of Coimbra

To my beloved parents
and
to my dear Amineh

Resumo

Nesta tese, lidamos o problema de estimar a estrutura 3D de superfícies não rígidas que sofrem deformação natural. Nós investigamos o problema com o objetivo de propormos idéias e soluções inovadoras que nos permitam relatarmos conclusões que contribuem para a área relevante. Portanto, o nosso foco está no desenvolvimento de algoritmos eficientes baseadas em formulações de problema que levam a modelos menos complexos e simultaneamente a resultados mais precisos. Enquanto a tese está concentrada em algoritmos especializadas para a reconstrução de superfícies não extensíveis ou extensíveis, no entanto, desenvolvemos o nosso trabalho para formulações generalizadas para que possa ser obtida uma modelagem consistente de ambos os tipos de deformação.

Todo o trabalho é um esforço organizado para realizar os objetivos da tese num programa de investigação de três tarefas, onde cada tarefa é dedicada a lidar com o problema de reconstrução com base numa nova idéia, usada para criar certos modelos de otimização em conformidade. O problema de reconstrução difere ligeiramente de cada tarefa para o próximo, dependendo de elementos como o tipo de câmera, o tipo de deformação, o modelo de deformação, a representação superficial, etc. Como resultado, três idéias gerais foram propostas para permitir reconstruções de superfície precisas.

Em resumo, cada idéia é fundado numa base particular, respectivamente, o uso da câmera de Tempo de Voo, combinação de restrições de deformação, modelagem de deformação a partir de transformações de superfície e buscamos estabelecer comparações confiáveis entre as três idéias com respeito a desempenho e aplicabilidade, avaliando os pontos fortes e fracos de suas algoritmos correspondentes. Para esse fim, um conjunto compreensivo de avaliações são realizadas tanto em sequências de vídeo reais e sintéticas que capturam diversas variantes de superfícies não rígidas.

Palavras-chave: Reconstrução, Superfícies Deformáveis, Técnicas de Otimização.

Abstract

In this thesis, we address the problem of estimating the 3D structure of non-rigid surfaces that undergo natural deformation. We thoroughly investigate the problem with the goal of proposing innovative ideas and solutions letting us report findings and conclusions that contribute to the relevant field. Therefore, our focus is on the development of efficient approaches based on problem formulations that lead to less complex models and simultaneously to more accurate results. While the thesis is concentrated on specialized approaches for the reconstruction of inextensible or extensible surfaces, we however develop our work toward generalized formulations so that consistent modeling of both deformation types can be derived.

The entire work is an organized effort to accomplish the thesis objectives in a 3-task research program where each task is dedicated to dealing with the reconstruction problem based on a novel idea, used to design certain optimization models accordingly. From each task to the next the reconstruction problem slightly differs depending on such elements as the camera type, deformation type, deformation model, the surface representation, etc. As a result, three general ideas have been proposed so as to enable accurate surface reconstructions by means of optimization.

In brief, each idea relies on a particular basis, respectively, the use of Time-of-Flight camera, combination of deformation constraints, deformation modeling from surface transformations and we seek to draw reliable comparisons between the three ideas with respect to performance and applicability, by evaluating the strengths and weaknesses of their corresponding approaches. For this purpose, a comprehensive set of experiments and evaluations is carried out on both real and synthetic video sequences capturing several variants of non-rigid surfaces.

Keywords: Reconstruction, Deformable Surfaces, Optimization-Based Techniques.

Acknowledgment

I would like to express special thanks to Professor Doctor Helder de Jesus Araujo for his supervision, support and encouragement. In addition, I am grateful to my friends and colleagues for their support and for their friendship as well. Lastly, I would like to thank ISR-UC for the facilities provided.

Contents

CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ALGORITHMS

CHAPTER 1

INTRODUCTION

In computer vision and computer graphics, 3D reconstruction is the process of capturing the shape and appearance of real objects. This process can be accomplished either by active or passive methods. If the model is allowed to change its shape in time, this is referred to as non-rigid or spatio-temporal reconstruction. The research of 3D reconstruction has always been a focus and difficulty. Using 3D reconstruction one can determine any object's 3D profile, as well as knowing the 3D coordinate of any point on the profile. The 3D reconstruction of objects is a generally scientific problem and core technology of a wide variety of fields, such as Computer Aided Geometric Design(CAGD), Computer Graphics, Computer Animation, Computer Vision, medical imaging, computational science, Virtual Reality, digital media, etc. For instance, the lesion information of the patients can be presented in 3D on the computer, which offers a new and accurate approach in diagnosis and thus has vital clinical value.

1.1 Motion Classification

The definitions of the different motion classes are briefly described as follows: Rigid motion is usually defined as a motion with no bending which preserves all distances and angles and has no associated non-rigidity. Articulated motion is piecewise rigid motion. It occurs in situations where individual rigid parts of an object move independently of one another. In this case, the motion of each constituent part is rigid, but the motion of the whole object is non-rigid. The rigid parts conform to the rigid motion constraints, but the overall motion is not rigid. We usually categorize it as articulated motion which has been a popular subject. Elastic motion is non-rigid motion whose only constraint is some degree of continuity or smoothness such as the motion of a heart, the waving of a cloth, or the bending of a metal sheet, where the shape of the object deforms under certain constraints. Fluid motion violates the continuity assumption. It may involve topological variations and turbulent deformations. Homothetic motion is motion with a uniform expansion or contraction of the surface. Conformal motion is non-rigid motion which preserves the angles between the curves on the surface, but not the distances. Quasi-rigid motion restricts the deformation to a small amount. A general motion is quasi- rigid when viewed in a sufficiently short interval of time, e.g. between image frames when the sampling rate is high enough. Isometric motion is defined as a motion which preserves distances along the surface as well as angles between curves on the surface. It can be described as a motion which preserves Gaussian but not mean curvature. The most well-known and simplest example of such motion is the bending of a plane surface into cylindrical shape. It is clear that any rigid motion is an isometric motion.

1.2 Single-View Reconstruction

The reconstruction of objects from a single image is under-constrained, meaning that the recovery of 3D shape is an inherently ambiguous problem. The case of non-rigid objects is even more complex and difficult [1, 2]. Given a specific configuration of points on the image plane, different 3D non-rigid shapes and camera motions can be found that fit the measurements.

Reconstruction approaches proposed over the past years can be categorized in two major types: those involving physics-based models [3, 4, 5, 6] and those relying on non- rigid structure-from-motion (NRSfM) approaches [7, 8, 9, 10, 11, 12, 13, 14]. In most cases, the former type ends up designing a complex objective function to be minimized over the solution space. The latter, on the other hand, takes advantage of prior knowledge on the shape and motion, to constrain the solution so that the inherent ambiguity can be tackled, and it performs effectively provided that the 2D point tracks are accurate and reliable. For example, Aanaes et al. [15] impose the prior knowledge that the reconstructed shape does not vary much from frame to frame while Del Bue et al. [16] impose the constraint that some of the points on the object be rigid.

1.3 Non-Rigid Structure From Motion

During the last years, several works have presented extensions of the Tomasi and Kanade's rigid structure from motion algorithm [17] - developed in the eraly 90' - to deal with the reconstruction of non-rigid objects [8, 18, 19, 20, 16, 21] - see Figure 1.1. These methods are based on the fact that any configuration of the shape can be described as a linear combination of basis shapes that define the principal modes of deformation of the object. In their pioneering work, Tomasi and Kanade have proposed the so-called factorization method for the 3D reconstruction. The key idea of this algorithm is the use of rank constraints to express the geometric invariants present in the data. This allows the factorization of a measurement matrix which contains the image coordinates of a set of features tracks into its shape and motion components. In the case of non-rigid factorization methods [11, 1, 2, 10, 7] which have been all originated from their rigid equivalent, the 3D shape recovered by the algorithms is represented as a linear combination of a number of detected modes of deformation. These models can be subsequently used as compact representations of the objects suitable for use in tracking [11], animation or other analysis. Bregler et al. [8] were the first to use a factorization method for the recovery of non-rigid structure and motion. His key insight was to use a low-rank shape model to represent a deforming shape as a linear combination of k basis shapes which encode the main modes of deformation of the object. Based on this model, they proposed a non-rigid factorization method for an affine camera that exploited the rank constraint on the measurement matrix and enforced orthonormality constraints on camera rotations to recover the motion and the non-rigid 3D shape. Different iterative optimization schemes were later introduced by Torresani et al. [21] and Brand [22] to improve the computation of the metric upgrade.

As an alternative to shape basis representation, Ijaz Akhter et al [23] propose a dual approach to describe the evolving 3D structure in trajectory space by a linear combination of basis trajectories. They describe a dual relationship between the two models (shape basis

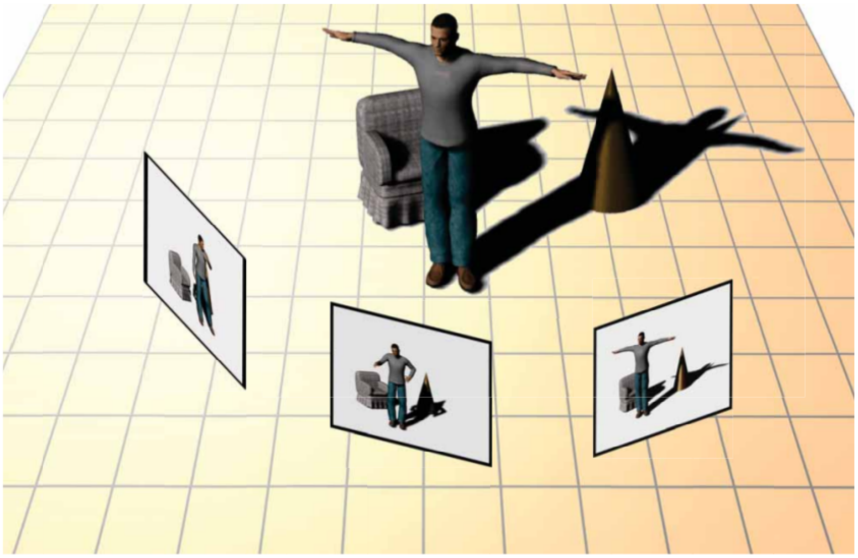

Figure 1.1: Structure from Motion.

and trajectory basis), showing that they both have equal power for representing 3D structure. Furthermore, they demonstrate that the principal advantage of expressing deforming 3D structure in trajectory space is that, in contrast to basis shapes, basis trajectories are object independent. This results in a significant reduction in unknowns, and corresponding stability in estimation. They propose the use of the Discrete Cosine Transform (DCT) as the object independent basis and empirically demonstrate that it approaches Principal Component Analysis (PCA) for natural motions. They report the performance of the proposed method, quantitatively using motion capture data, and qualitatively on several video sequences exhibiting nonrigid motions including piecewise rigid motion, partially nonrigid motion (such as a facial expressions), and highly nonrigid motion (such as a person walking or dancing).

Most NRSfM methods perform under affine projection. However, these algorithms fail to give reliable 3D shape estimates when the overall rigid motion of the sequence is small. To cope with this limitation, Xavier Lladó [24] propose a RANSAC-based approach for the 3D Euclidean reconstruction of deformable objects observed by an uncalibrated stereo rig. They prove that using a stereo setup drastically improves the 3D model estimation when the observed 3D shape is mostly deforming without undergoing strong rigid motion. Also, they consider the case where images are acquired at closer distances, with a wide field of view or the scene is large in space. Also, there are NRSfM algorithms that work under the full perspective camera case [11, 7, 25, 26]. The main constraint of these algorithms is that a reliable model can only be extracted if the image sequence includes a large rotation component.

The use of the dimensionality of the subspace in which the image trajectories lie, has been commonly considered to perform motion segmentation [27, 28]. However, all these methods focus either on the segmentation of independently moving objects or on the segmentation of objects of different nature (rigid, non-rigid), but none of them can deal efficiently with the segmentation of rigid and non-rigid points on a single deformable object. Moreover, most of these methods assume an affine camera model Del Bue et al. [29] evaluate a method that performs the automatic segmentation of a set of rigidly moving points within a deformable object given a set of 2D image measurements and study a full perspective camera case. Their work is the first attempt to obtain a reliable segmentation of rigid points from non-rigid bodies.

1.4 Priors and Constraints

Although the low-rank shape model has proved to be a successful representation, however the non-rigid structure from motion problem is under-constrained. Xiao et al. [12] proved that the orthogonality constraints were insufficient to disambiguate rigid motion and deformations. They identified a new set of constraints on the shape bases which, when used in addition to the rotation constraints, provide a closed form solution to the problem of non-rigid structure from motion. However, their solution requires that there be K frames (where K is the number of basis shapes) in which the shapes are known to be independent. Non-linear optimization schemes that minimize image reprojection error have also been proposed to refine an initial solution [30]. The advantage of these methods is that they provide a maximum likelihood estimate in the presence of Gaussian noise and prior knowledge on any of the model parameters can be easily incorporated into the cost function in the form of penalty terms. The need for incorporating prior information on the motion or shape parameters to avoid the ambiguities inherent to non-rigid shape estimation is also recognized by Torresani et.al. [21, 19] who propose an algorithm that learns the time-varying shape of a non-rigid 3D object from uncalibrated 2D tracking data. Temporal smoothness in the object shape can be imposed within their framework which can also handle missing data. Aanaes et al. [15] impose the prior knowledge that the reconstructed shape does not vary much from frame to frame, while Del Bue et al. [16] impose the constraint that some of the points on the object are rigid. Both approaches use bundle adjustment to refine all the parameters of the model together. Bartoli et al. [20] on the other hand, use a coarse to fine shape model where new deformation modes are added iteratively to capture as much of the variance left unexplained by previous modes as possible.

The priors can be divided in two main categories: the statistical and physical priors. For instance, the methods relying on the low-rank factorization paradigm [15, 16] can be classified as statistical approaches. Learning approaches such as [31, 32, 33] also belong to the statistical approaches. Physical constraints include spatial and temporal priors on the surface to be reconstructed [34, 35].

1.5 Deformable Surface Reconstruction

Monocular reconstruction of deformable surfaces has been extensively studied in the last few years [36, 37]. Strictly-speaking, isometric reconstruction from perspective camera views has attracted much of the attention. A physical prior of particular interest in this case is the hypothesis of having an inextensible (i.e. isometric) surface [38, 39, 40, 41]. This hypothesis means that the length of the geodesics between every two points on the surface should not change across time, which makes sense for many types of material such as paper and some types of fabric.

The reconstruction of deformable surfaces is becoming increasingly important and this can be visible considering its practical applications. Physics has inspired early approaches. These approaches amount to a minimization based on the physical behaviour of the surface [3, 42, 43]. Although it makes sense that we integrate physical laws into our algorithms, the final framework will be affected by two major shortcomings:

- The material parameters, which are typically unknown, have to be determined.
- In order to estimate the parameters accurately, in the presence of large deformations, we need to build a complex cost functional (which is hard to optimize).

Methods that learn models from training data were introduced to overcome these limitations. In this case, surface deformations are expressed as linear combinations of deformation modes which are obtained from training data. NRSfM methods built on this principle recover simultaneously the shape and the modes from image sequences [11, 21, 7]. Although this is a very attractive idea, practical implementations are not easy since they require points to be tracked across the entire sequence. Moreover, they are only effective for relatively small deformations. There have also been a number of attempts at performing 3D surface reconstruction without using a deformation model. One approach is to use lighting information in addition to texture clues to constrain the reconstruction process [44, 45], which has only been demonstrated under very restrictive assumptions on lighting conditions and is therefore not generally applicable. The algorithms for reconstructing deformable surfaces can be classified by the type of the surface model (or representation) used: Point-wise methods only reconstruct the 3D position of a relatively small number of feature points, resulting in a sparse reconstruction of the 3D surface [39]. Physics-based models such as superquadrics [43], triangular meshes [40] or Thin-Plate Splines (TPS) [39]

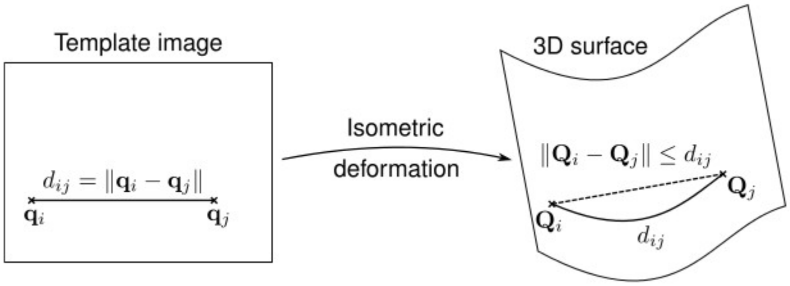

Figure 1.2: Upper-bound model.

have been also utilized in other algorithms. In TPS, the 3D surface is represented as a parametric 2D-3D map between the template image space and the 3D space. Then, a parametric model is fit to a sparse set of reconstructed 3D points in order to obtain a smooth surface which is not actually used in the 3D reconstruction process.

Having an isometric surface means that the length of the geodesics between pairs of points remains unchanged when the surface deforms and the deformed surface can be obtained by applying an isometric transformation (map) to a template surface. In many cases, computation of the geodesics is not trivial and involves the application of differential geometry. Instead, the Euclidean distance, which is much easier to estimate, has been regarded as a good approximation to the geodesic distance, on condition that it does not drop too much below the geodesics. Euclidean approximation is better when there are a large number of points. Although it can work well in some cases, it gives poor results when creases appear in the 3D surface. In this case, the Euclidean distance between two points on the surface can shrink. For this reason, the ‘upper bound approach’ has been proposed which relies on the fact that the Euclidean distance between 2 random points on a plane is necessarily less than (or equal to) the corresponding geodesics, which is known as inextensibility constraint. As a result, early approaches relax the non-convex isometric constraints to inextensibility with the so-called maximum depth heuristic [39, 37, 46]. The idea is to maximize point depths so that the Euclidean distance between every pair of points is upper bounded by its geodesic distance, computed in the template [33, 40]. This constraint is called *upper-*

bound model - see Figure 2.2. In these papers, a convex cost function combining the depth of the reconstructed points and the negative of the reprojection error is maximized while enforcing the inequality constraints arising from the surface inextensibility. The resulting formulation can be easily turned into a SOCP problem. This problem is convex and gives accurate reconstructions. A similar approach is explored in [38]. The approach of [39] is a point-wise method. The approaches of [33, 40, 38] use a triangular mesh as surface model, and the inextensibility constraints are applied to the vertices of the mesh. Recently, analytical solutions for isometric and conformal deformations have been provided by posing them as a system of Partial Differential Equations [47, 48, 49]. The approach was developed under weak-perspective projection and requires complex differential models. A SLAM method for elastic surfaces was developed, using fixed boundary conditions [50]. In [51], the authors formulate the reconstruction problem of a generic surface in terms of the minimization of stretching energy and impose a set of fixed boundary 3D points to constrain the solution. This approach deals with a general group of elastic surfaces without applying any constraints explicitly associated with conformal deformation.

1.6 Publications

Most of the thesis is based on the following publications and achievements:

1.6.1 Peer-reviewed journal articles

- S. Jafar Hosseini, Helder Araújo: From D-RGB Based Reconstruction Toward a Mesh Deformation Model for Monocular Reconstruction of Isometric Surfaces, EURASIP Journal on Image and Video processing, 2016(1), 1-11, (doi:10.1186/s13640-0160114-9).
- S. Jafar Hosseini, Helder Araújo: SDP-Based Approach to Monocular Reconstruc-

tion of Inextensible Surfaces, Journal of IET, Computer Vision, 2016, Volume 11, Issue 1, 43-49, (doi:10.1049/iet-cvi.2016.0006).

- S. Jafar Hosseini, Helder Araújo: Patch-Based Reconstruction of Surfaces Undergoing Different Types of Deformations, Journal of Signal, Image and Video Processing, March 2017, 1–8, (doi:10.1007/s11760-017-1079-6).

1.6.2 Peer-reviewed international conference papers

- S. Jafar Hosseini, Helder Araújo: Structure from Motion - ToF-aided 3D Reconstruction of Isometric Surfaces, ICPRAM 2014, 544-552. [A ToF-Aided Approach to 3D Mesh-Based Reconstruction of Isometric Surfaces. ICPRAM (Selected Papers) 2014, 146-161]
- S. Jafar Hosseini, Helder Araújo: 3D Estimation of Isometric Surfaces Using a ToF-Based Approach, ISVC 2014, 129-140.
- S. Jafar Hosseini, Helder Aráujo: 3D Estimation of Extensible Surfaces Through a Local Monocular Reconstruction Technique, ACCV Workshopon 3D Modelling and Applications, 2016, 114-123.
- S. Jafar Hosseini, Helder Araújo: Monocular Reconstruction of Conformally Distorted Surfaces, In proceedings of EUVIP 2016.

CHAPTER 2

MOTIVATIONS, BASIC IDEAS AND CONCEPTS

Figure 2.1: Inextensible Deformation.

3D reconstruction is a classic computer vision topic which has been explored carefully over the years. Traditionally, this topic coupled with camera calibration forms the core of numerous practical applications ranging from security surveillance to medical inspections. Therefore, 3D reconstruction systems are treated as visual perception tools that replace human eyes specially when speed and accuracy are crucial factors in some application.

3D reconstruction deals with the recovery of structures which could be either still or in motion. Rotational or translational motion does not alter the shape of an object, whereas other types of motion involve a change in relative positions of parts of the object surface. In the latter case, such a change is called *deformation*.

Deformations vary according to how points on the object move in relation to one another. They fall into two general groups: when a deforming structure undergoes no extensions or contractions, the deformation is referred to as inextensible or isometric (see Figure 2.1) but in case of extensions or contractions, it is called extensible (see Figure 2.2). We address both types of deformations in this dissertation.

The entire work carried out till the completion of the expected experiments is split

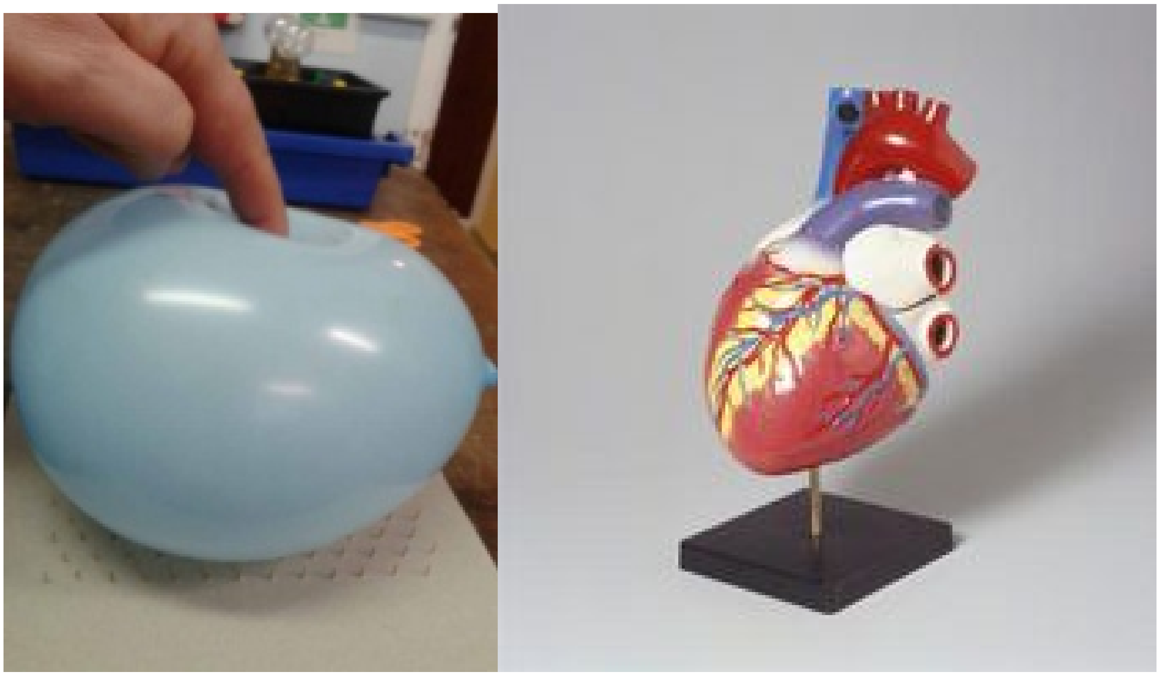

Figure 2.2: Extensible Deformation.

into three major tasks depending on the basic elements that determine the input to the reconstruction problem. Examples of these elements are the camera type, deformation type, deformation model and the model considered to represent the surface. In other words, here the reconstruction problem can take several inputs but comes with a particular unknown output (that is 3D structure) and we attempt to solve for the same unknown in different ways. The main goal of this work is to make significant contributions to the context of deformable surface reconstruction. For this purpose, we closely investigate the problem and propose innovative ideas and solutions. More precisely stated, we tackle the reconstruction problem aiming to obtain and report findings and conclusions that positively affect the relevant field. In the following, we give a brief description of the three tasks. The details of these tasks will be later discussed and presented in a separate chapter for each task.

2.1 D-RGB Based Reconstruction: Mesh Representation

The advent of active range cameras has introduced a new research direction in 3D perception, resulting in highly efficient reconstruction systems. Although the accuracy of 3D measurements produced by early range cameras was not promising, however the evolving technology of such cameras has been in rapid progress in the last decade and today the emergence of such cameras as Kinect has fundamentally influenced applications that make use of 3D data.

Time-of-Flight (ToF) cameras are a common sort of range cameras which deliver a depth map of the scene in real time - see Figure 2.3. Most ToF cameras perform with relatively acceptable accuracy. However errors might occur due to the conditions under which these cameras operate. We also find it interesting to test a ToF camera and try to use it for deformable surface reconstruction. For this purpose, we define a reconstruction problem and develop an appropriate algorithm to solve it using the ToF camera.

2.1.1 Motivation and Problem Statement

Linear deformation model is commonly applied when estimating isometric deformations as a 3D mesh. In order to learn this model, it is necessary that prior data of a sufficient variety of possible isometric deformations of the mesh be available in advance. Because of its material, flexibility and other physical properties, every object that undergoes isometric deformation might show a limited class of deformations, meaning that it is possible that two isometric surfaces of different material do not share alike deformations. Therefore, the linear deformation model is object-specific and should be learned for each different object independently of others. The objective of this task is to complete the linear deformation model given the following elements:

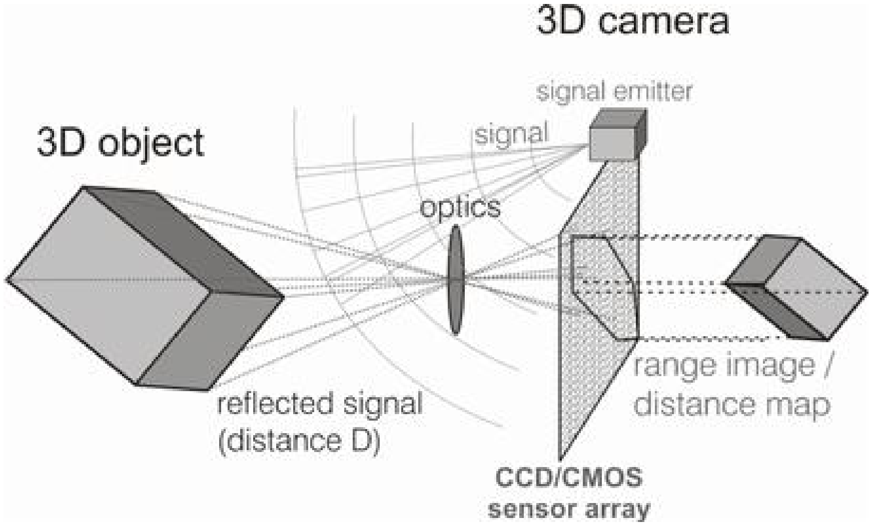

Figure 2.3: ToF sensor principle.

- Surface representation: a triangular mesh is assumed to cover the surface. This mesh consists of edges and vertices.
- Primary output: the coefficients of the linear deformation model.
- Secondary output (i.e. 3D structure): the 3D positions of the vertices at a single frame.
- Available data: a flat 3D template of the mesh and 2D images of a set of feature points distributed over the mesh and tracked on a single RGB image.
- Deformation type: isometric.
- Camera setup: a ToF camera is applied along with a conventional camera.

2.1.2 Solution Summary

After applying registration between the two cameras, the depth of the feature points is computed. Next step is to estimate the depth of the vertices. For this, a linear programming (LP) problem is defined based on the available depth of the feature points. The x and y

coordinates of the vertices are still unknown and remain to be determined. Hence, a second order cone programming problem (SOCP) is designed to obtain the 3D positions of the vertices. This optimization is based on an adaptation of the upper-bound model which actually serves as a constraint imposed by the type of deformation under consideration. At this point, the 3D structure of the mesh (our secondary output) becomes available. This algorithm is used with several different deformations in order to acquire the data required to learn the deformation model. In the end, Principal Component Analysis is applied to this data to complete the deformation model by calculating its coefficients.

Beside, we will examine the complete deformation model by employing one of the traditional methods for the reconstruction of isometric surfaces.

2.2 Monocular Reconstruction: Combination of Deformation Constraints

In order to formulate a reconstruction problem for a deformable surface, the type of deformation considered for reconstruction should be expressed using standard mathematics. As a result, each deformation type is characterized by a specific mathematical constraint. Therefore, extensible and inextensible deformations differ according to their deformation constraint. The deformation constraints that have already been defined for inextensible surfaces include those based on the upper-bound model and those derived from the differential properties of such surfaces. Other than inextensible surfaces, in this task we also deal with a particular class of extensible deformations, that is conformal deformation. The two sets of constraints mentioned above shall be slightly different for conformal surfaces. For example, an extended form of the upper-bound model may apply to these surfaces.

In addition to linear deformation model, interpolation is a means of modeling deformations based on an initial fixed set of feature points. Generally speaking, warping functions such as Free Form Deformations (FFD) enables mapping from image space to 3D space i.e. image points to 3D points on the surface, thereby leading to smooth reconstructions. Such mapping approximately covers every natural deformation and facilitates the use of constraints associated with differential properties.

2.2.1 Motivation and Problem Statement

To the best of my knowledge, no previous work has addressed the joint application of an upper-bound model-based constraint and a differential-based constraint. It is quite obvious that the simultaneous use of both constraints will lead to improved results compared with the use of either of them alone. The goal of this task is to reformulate these constraints in a unique way so that they can be integrated into a single reconstruction problem which is made up of the following elements:

- Surface representation: Instead of using the mesh representation, here the surface is represented as a distribution of a number of feature points across the surface. As a matter of fact, the smooth 3D shape of the surface is identified by these feature points accompanied by an extremely large number of points interpolated by the use of FFD.
- Output (i.e. 3D structure): the 3D positions of the feature points.
- Available data: a flat 3D template of the surface with the positions of the feature points available. Also their image points tracked on a single image are given.
- Deformation type: isometric, conformal.
- Deformation model: a FFD warp is employed to model the deformations.
- Camera setup: a perspective camera.

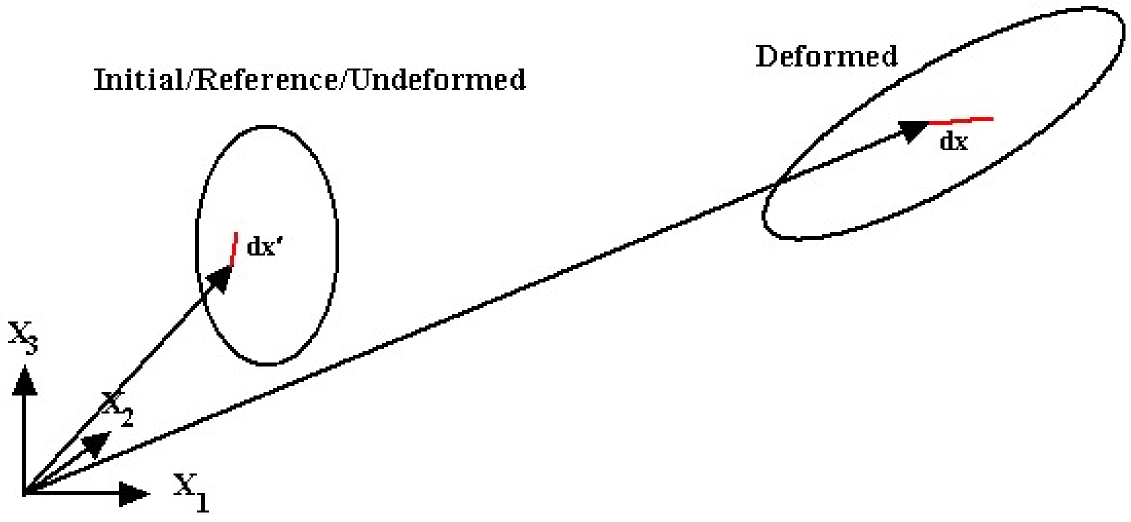

Figure 2.4: Surface transformation.

2.2.2 Solution Summary

In case of inextensible surfaces, we formulate the reconstruction problem as a semi-definite programming (SDP) problem where a combination of the upper-bound model and a differential-based constraint is used as deformation constraints.

In case of conformal surfaces, a sequential second-order cone programming is defined in order to estimate the 3D positions of the feature points. A modified upper-bound model together with some sort of differential-based constraints constitutes deformation constraints in this case.

In both cases, the 3D template is used for setting the parameters of the deformation constraints.

2.3 Monocular Reconstruction: Surface Transformations

As illustrated in Figure 2.4, the surface shape undergoes a smooth change as the surface deforms. The change of shape between two instants can provide an insight into how to model the deformation regardless of its type. One possibility is to attempt to express this physical change as a mathematical transformation that relates the initial and the deformed shape.

Note that this transformation acts on a set of points spread on the surface and allows the computation of their positions on the deformed shape, given their initial positions. Thus, the surface representation in this case is point-wise. Also depending on the type of deformation, such a transformation may take various forms.

2.3.1 Motivation and Problem Statement

In the context of deformable surface reconstruction, the use of a surface transformation to model deformations has not received great attention knowing that it is potentially helpful. In this work, that strategy has been of particular interest so that efficient reconstruction methodologies might be devised. Therefore, the main goal of this task is to propose an algorithm based on a generalized transformation that is applicable to a wide range of deformations, including inextensible and extensible ones. This algorithm should perform effectively for any natural deformation type. In this case, the reconstruction problem is as follows:

- Surface representation: The surface is represented as a distribution of a number of feature points across the surface. In fact, we use a point-wise representation.
- Primary output (i.e. 3D structure): the 3D positions of the feature points.
- Secondary output: the parameters of the transformation.
- Available data: a 3D initial shape of the surface with the positions of the feature points available. Also their image points tracked on a single image are given.
- Deformation type: isometric, extensible.

- Deformation model: a surface transformation between the initial and the deformed shapes.
- Camera setup: a perspective camera.

2.3.2 Solution Summary

The observation that any natural deformable surface shows piece-wise homogeneous deformation has been ignored in the respective area. This observation implies that the total deformation of a natural surface is composed of local homogeneous deformations. Therefore, the probability that a surface part undergoes such deformation is in inverse relationship with its size, meaning that smaller surface parts are more likely to show homogeneous deformation. For this reason, the surface is divided into several patches that do not overlap but together cover the entire area of the surface.

Homogeneous deformation is simply characterized by a linear surface transformation between the initial shape and the deformed shape of the surface. To determine the parameters of this transformation, we define an optimization scheme which is applied to each patch separately. Once known, the transformation can be used to estimate the 3D positions of the points on the deformed patch.

Since the transformation depends on the initial shape to be given before, we also propose a rigid reconstruction technique based on Homography in order to obtain the 3D positions of the points on the initial shape.

The algorithm described above can be used for both extensible and inextensible deformations. However, we also propose a separate algorithm for extensible surfaces by addressing a specific type of extensible deformation called homothetic. This refers to uniformly extensible deformation. Considering that any extensible deformation comprises local homothetic deformations, again we propose a patch-based reconstruction method by means of a surface transformation between the initial and the deformed shape. Note that this second algorithm is only intended to reconstruct extensible surfaces. Since homothetic deformation is a particular subset of homogeneous deformation, the homothetic transformation is a subspace of the homogeneous transformation.

CHAPTER 3

D-RGB BASED RECONSTRUCTION WITH A 3D MESH

3.1 Introduction

In this chapter, I deal with the first task in detail and study 3D reconstruction of surfaces deforming isometrically. Given that an isometric surface is represented by means of a triangular mesh and that feature/point correspondences on an image are available, the goal is to estimate the 3D positions of the mesh vertices. To perform such monocular reconstruction, a common practice is to adopt *linear deformation model*. I also integrate this model into a least-squares optimization. However, this model is obtained through a learning process requiring an adequate data set of possible mesh deformations. Providing this prior data is the primary goal of this work and therefore a novel reconstruction technique is proposed for a mesh overlaid across a typical isometric surface. This technique consists in the use of a range camera accompanied by a conventional camera and implements the path from the depth of the feature points to the 3D positions of the vertices through convex programming. The idea is to use the high-resolution images from the RGB camera in combination with the low-resolution depth map to enhance mesh deformation estimation. With this approach, multiple deformations of the mesh are recovered with the possibility that the resulting deformation model is simply extended to any other isometric surfaces for monocular reconstruction. Experimental results show that the proposed approach is robust to noise and generates accurate reconstructions.

3.1.1 Problem Formulation

We aim at the reconstruction of surfaces that undergo isometric deformations. Assuming that a triangular mesh is used to represent an isometric surface and that a set of feature/point correspondences on an image of the surface have been provided, the objective is to determine the 3D positions of the mesh vertices. To carry out this monocular reconstruction, we formulate a nonlinear least squares optimization that integrates the linear deformation model, deformation-based constraints which we call *isometric constraints* and the projection equations in order to solve for 3D positions of the mesh vertices.

Main Contribution: Several reconstruction methods have previously relied on the linear deformation model as a crucial element that can reduce the ambiguity of infinite solutions. This model is specially useful when using the mesh representation. It is typically obtained from prior training data that corresponds to various possible deformations of the mesh. As a result, it is required to reconstruct these mesh deformations beforehand, which is challenging without some sort of supporting 3D information. Furthermore, the precision of the training data is important and must be ensured. For this purpose I propose an innovative technique to acquiring such data with high accuracy. This technique aims to estimate a regular 3D mesh overlaid across a generic isometric surface and is used to recover several different deformations of the mesh in a way that makes it possible to extend the computed deformation model to other isometric surfaces for monocular reconstruction. In developing this approach, I use a conventional RGB camera aided by a range camera. My emphasis is, in fact, on the use of a Time-of-Flight (ToF) camera in conjunction with the RGB camera. Most RGB cameras provide high-resolution images. With these cameras, one can use efficient algorithms to calculate the depth of the scene, recover object shape or reveal structure, but at a high computational cost. ToF cameras deliver a depth map of the scene in real-time but with insufficient resolution for some applications. So, a combination of a conventional camera and a ToF camera can exploit the capabilities of both. We assume that the fields of view of both cameras mostly overlap. From the depth map, the depth of the feature points can be extracted by adopting a registration technique for the camera combination. This allows the depth of the mesh vertices to be subsequently computed using either a linear system of equations or a linear programming problem. Given the mesh depth data, the complete 3D positions of the vertices can be recovered through a second-order cone programming (SOCP) problem. Applying the approach just described to a variety of mesh deformations leads to the required data, thereby computing the deformation model.

Organization of this chapter: This chapter is organized as follows: the first section discusses the background of my work, including the mesh representation and the linear deformation model. Next, the monocular reconstruction is explained. The following section is assigned to a detailed description of my D-RGB based reconstruction. Then, experimental results and quantitative evaluations are presented, demonstrating the efficiency of my reconstruction schemes. Finally, I discuss conclusions.

3.2 Background

3.2.1 Mesh Representation

Assume that a set of 3D feature points $\mathbf{p}^{ref} = \left\{ \mathbf{p}_1^{ref}, \ \cdots, \ \mathbf{p}_N^{ref} \right\}$ on a template with a known shape (usually a flat surface), and a set of 2D image points $\mathbf{q} = \left\{ \mathbf{q}_1, \ \cdots, \ \mathbf{q}_N \right\}$ tracked on the RGB input image of the same surface, but with a different and unknown deformation are given. As already stated, we represent the surface as a triangular 3D mesh with n_v vertices $\mathbf{v}_i$ (and n_{tr} triangles) concatenated in a vector $\mathbf{s} = \left[\mathbf{v}_1^T, \ \cdots, \ \mathbf{v}_{n_v}^T \right]^T$, and denote by $\mathbf{s}^{ref}$ the template mesh, and $\mathbf{s}$ the mesh we seek to estimate - see Figure 3.1. Let $\mathbf{p}_i$ be a feature point on the mesh $\mathbf{s}$ corresponding to the point $\mathbf{p}_i^{ref}$ in the template. We can express $\mathbf{p}_i$ in terms of the barycentric coordinates of the triangle it belongs to:

$$\mathbf{p}_i = \sum_{j=1}^{3} a_{ij} \mathbf{v}_j^{[i]} \tag{3.1}$$

where the a_{ij} are the barycentric coordinates and $\mathbf{v}_j^{[i]}$ are the vertices of the triangle containing the point $\mathbf{p}_i$. Mesh representation has the advantage of simplifying reconstructions in view of the fact that the isometric type of deformation imposes the constraint that the length of the edges of a mesh with a dense distribution of vertices stay nearly the same, as the surface deforms. As a result, we may treat the mesh triangles as rigid,

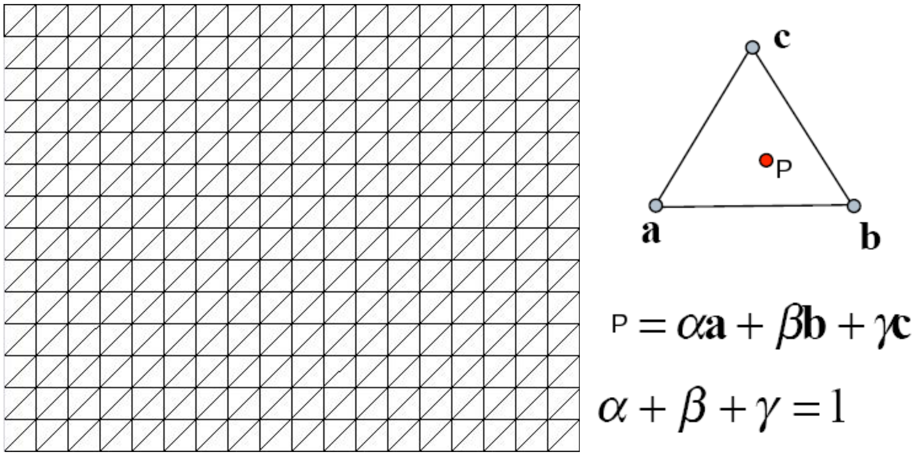

Figure 3.1: A regular triangular mesh and barycentric coordinates.

allowing us to consider that the barycentric coordinates remain constant for each point. These coordinates are easily computed from points $\mathbf{p}_i^{ref}$ and the mesh $\mathbf{s}^{ref}$. Let us denote by $\mathbf{A} = \left\{ \mathbf{a}_1, \ \cdots, \ \mathbf{a}_N \right\}$ the set of barycentric coordinates associated with the feature points, where $\mathbf{a}_i = \left[\begin{array}{ccc} a_{i1}, & a_{i2}, & a_{i3} \end{array} \right]$.

3.2.2 Linear Deformation Model

The space of possible deformed shapes of the surface is constrained by applying a deformation model. This model adequately fills in the missing information while being flexible enough to allow reconstruction of complex deformations [32]. A mesh deformation is thus modelled as a linear combination of a mean shape $\mathbf{s}_0$ and n_m basis shapes (deformation modes) $\mathbf{S} = [\mathbf{s}_1, ..., \mathbf{s}_{n_m}]$:

$$\mathbf{s} = \mathbf{s}_0 + \sum_{k=1}^{n_m} w_k \mathbf{s}_k = \mathbf{s}_0 + \mathbf{S}\mathbf{w} \tag{3.2}$$

These modes can be obtained by applying Principal Component Analysis (PCA) to a plenary set of training deformations. In my work, this training data is acquired using a high-resolution image combined with the knowledge of the depth of a set of feature points.

3.3 Monocular Reconstruction from a Single View

Given that the linear deformation model has been computed, the objective is to proceed with an efficient algorithm which is intended to demonstrate the use of the linear deformation model in monocular reconstruction of mesh deformations. For this purpose, I introduce an algorithm that falls within a particular class of methods which follow the same basic principle, namely, mesh representation along with linear deformation model [32, 33, 37, 40]. My algorithm is slightly different, which is composed of two non-linear constraints. It is capable of performing such reconstruction that the shape of any isometrically deformed surface is estimated by using only a conventional camera.

Isometric Constraint: This constraint is the difference between the observed and the predicted length of an edge. We formulate a non-linear constraint as:

$$e_{iso} = \sum_{i=1}^{n_e} \left(L_i - \left\| \mathbf{s}_1^{[i]} - \mathbf{s}_2^{[i]} \right\| \right)^2 \tag{3.3}$$

where L_i is the length of the edge i, computed on the template. $\mathbf{s}_1^{[i]}$ and $\mathbf{s}_2^{[i]}$ denote the two entries of the mesh that account for the ending vertices of the edge i.

Reprojection Error: In addition, there are also reprojection errors, that is, errors on the image position of the feature points. We should thus account for the reprojection error by adding a term to the function to be optimized. By combining Equations 3.1 and 3.2, we will have:

$$\mathbf{p}_i = \sum_{j=1}^{3} a_{ij} \mathbf{s}_j^{[i]} = \sum_{j=1}^{3} \left(\mathbf{s}_{0j}^{[i]} + \mathbf{S}_j^{[i]} \right) \tag{3.4}$$

where $\mathbf{s}_{0j}^{[i]}$ and $\mathbf{S}_j^{[i]}$ are the subvector of $\mathbf{s}_0$ and the submatrix of $\mathbf{S}$ (respectively), corresponding to the vertex j of the triangle in which the feature point i resides. The term corresponding to the reprojection error can be obtained as indicated below.

$$e_{re} = \sum_{i=1}^{N} \left\| \breve{}\,\mathbf{i} \begin{bmatrix} \mathbf{q}_i \\ 1 \end{bmatrix} - \left[\mathbf{K}^{\circ}_{rgb} . \mathbf{p}_i \right] \right\|^2 \tag{3.5}$$

where N is the number of feature points. The λ_i add extra unknowns to the optimization problem. Therefore, it is advantageous to reformulate the above equations so that the λ_i can be eliminated. Consider the equation below:

$$\lambda_i \begin{bmatrix} \mathbf{q}_i \\ 1 \end{bmatrix} = \mathbf{K}^{\circ}_{rgb} \left[\sum_{j=1}^{3} a_{ij} \mathbf{s}_j^{[i]} \right] \tag{3.6}$$

After some simple algebraic manipulation, we obtain:

$$\begin{bmatrix} a_{i1}\mathbf{A}_i & a_{i2}\mathbf{A}_i & a_{i3}\mathbf{A}_i \end{bmatrix}_{2\times 9} \begin{bmatrix} \mathbf{s}_1^{[i]} \\ \mathbf{s}_2^{[i]} \\ \mathbf{s}_3^{[i]} \end{bmatrix}_{9\times 1} =$$

$$\begin{bmatrix} e_{1,i} \\ e_{2,i} \end{bmatrix}_{2\times 1} = 0 \;\text{ where } \mathbf{A}_i = \mathbf{K}^{\circ(1:2)}_{rgb} - \mathbf{q}_i \mathbf{K}^{\circ(3)}_{rgb} \tag{3.7}$$

This equation provides 2 linear constraints as: $e_{1,i} = 0$ and $e_{2,i} = 0$. Thus, the modified e_{re} takes a form where the λ_i does not exist, as follows:

$$e_{mre} = \sum_{i=1}^{N} \left((e_{1,i})^2 + (e_{2,i})^2 \right) \tag{3.8}$$

where e_{mre} denotes the modified e_{re}.

Objective Function: We have now derived two constraints, described as two separate non-linear expressions. However, I intend to integrate both constraints into a single objective function so that they are taken into account at one time, while estimating all the parameters. To do so, we minimize the weighted summation of them in such a way that the reprojection error term is assigned a weight *m* that accounts for its relative influence within the combined objective function.

$$\min_{\mathbf{W}} e_{tot} = (e_{iso} + m.e_{mre}) \tag{3.9}$$

The above optimization scheme is a non-linear least-squares minimization problem, typically solved using an iterative algorithm such as Levenberg-Marquardt.

3.4 Reconstruction Using a D-RGB Camera Setup

In order to build an adequate data set of mesh deformations for learning the deformation model, I propose a reconstruction approach for a typical surface based on a D-RGB camera setup. The completed deformation model can be then extended for monocular reconstruction of any other surfaces that undergo isometric deformations. Using the result of the registration described below, we can obtain an estimate for the depth of the feature points. The idea behind my D-RGB based reconstruction is to determine the 3D positions of the mesh vertices, given this depth data. This is done in 2 steps: first the depth of the vertices is estimated and then their *xy*-coordinates.

Registration Between Depth and RGB Images: The resolutions of the depth and RGB images are different. A major issue that directly arises from the difference in resolution is that a pixel-to-pixel correspondence between the two images can not be established even if the FOVs fully overlap. Therefore the two images have to be registered so that a mapping between the pixels in the depth image and in the RGB image can be established.

The depth images provided by the ToF camera are sparse and affected by errors. Several methods can be used to improve their resolution [52, 53, 54, 55], allowing the estimation of dense depth images. However, to estimate depth for all the pixels of the RGB image, based on the depth map given by the ToF camera, simple linear procedures are used as follows:

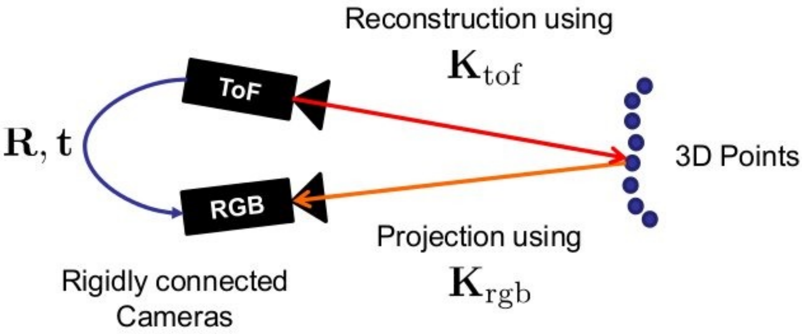

Figure 3.2: ToF/RGB camera setup.

I use a pinhole camera model for both cameras and assume that they are calibrated internally and that also the relative pose between both cameras, specified by the rotation matrix $\mathbf{R}^{'}$ and translation vector $\mathbf{t}^{'}$ has been estimated. Let $\mathbf{p}_{tof}$ and $\mathbf{p}_{rgb}$ represent the 3D positions of a point in the coordinate system of the Tof and the RGB cameras, respectively. $\mathbf{p}_{tof}$ is obtained directly from the calibrated ToF camera. Thus $\mathbf{p}_{rgb}$ can be easily calculated by: $\mathbf{p}_{rgb} = \mathbf{R}^{'}\mathbf{p}_{tof} + \mathbf{t}^{'}$ - see Figure 3.2. For each feature point on the RGB image, we select the 4 closest neighbours whose depth was obtained from the depth image. Then, a bilinear interpolation is performed. Another possibility could be to select the 3 closest neighboring pixels (therefore, defining a triangle) and assume that the corresponding 3D points define a plane. An estimate for the depth of the feature point could be then obtained by intersecting its projecting ray with the 3D plane defined by the three 3D points. As a result, the depth of the N feature points is computed accurately and we indicate by $p_{z,k}$ the depth of the feature point k.

3.4.1 Step 1: Recovery of the depth of the vertices

Given $p_{z,k}$ for all ks, the goal is to estimate the depth of the vertices. Let z_i and rz_j denote the depth of the vertex i and the relative depth of the edge j, respectively. The vertices are numbered and sorted according to a particular ordering. The same goes for the set of all relative depths. In addition, a relative depth needs to conform to either of the two directions along its edge, i.e. $rz_{25} = z_{16} - z_7$ or vice-versa. So, a pre-defined set of selected directions is applied to all edges. As a matter of fact, the rigidity of a closed triangle enforces the fact that the sum of the depth differences between every 2 vertices concatenated around the triangle, be zero. This can be expressed with relative depths and gives us n_{tr} equations which, in conjunction with the equations associating the relative depths with the depth of vertices, add up to $n_{tr} + n_e$ (the number of triangles + the number of edges) linear equations. We augment this linear system with the depth of the feature points. From Equation 3.1, we can derive $p_{z,i} = a_{i1}z_1^{[i]} + a_{i2}z_2^{[i]} + a_{i3}z_3^{[i]}$. Having this equation for every feature point results in N linear independent equations. Putting together all the equations available, we end up with $n_{tot} = n_{tr} + n_e + N$ linear equations where the only unknowns are the depth of vertices and of the edges (i.e. $n_v + n_e$ unknowns), which means that the resulting linear system is overdetermined. We denote this linear system as $\mathbf{Mx} = \begin{bmatrix} \mathbf{0} \\ \mathbf{p}_z \end{bmatrix}$. I now propose two algorithms for determining the depth of the mesh vertices below.

Algorithm 1: Solving A Linear System of Equations: The linear system above has $n_e + N$ independent equations out of n_{tot} and this is not yet enough to find the right single solution because there are still an infinitude of solutions that satisfy this linear system. One possible alternative to handle this is to fit an initial mesh using polynomial interpolation, to the data. This fitting consists in xy-coordinates of the feature points on the template as input and their z-coordinates on the input deformation as output. Once the parameters of the interpolant have been found, we can obtain initial estimates for depth of the vertices, with their xy-coordinates on the template as input. Let $z_i^{\prime}$ be the interpolated depth of the vertex i. By adding this result as an extra equation to the linear system described earlier, we obtain the modified linear system $\mathbf{M}_{mod}\mathbf{x} = \mathbf{b}$, which has most likely full column-rank. So, the number of independent equations out of $n_{tot} + n_v$ will be $n_e + n_v$. Since the number of independent equations is equal to that of unknowns, there must be a unique solution which can be computed via the normal equations. In general, the use of least-squares minimization leads to better results.

Algorithm 2: A Linear Programming Problem: An LP can be also defined to estimate the depth of the vertices. The linear system $\mathbf{Mx} = \begin{bmatrix} \mathbf{0} \\ \mathbf{p}_z \end{bmatrix}$ is used as a set of constraints in this LP. However, it is essentially useful to have additional estimates for the depth of the mesh vertices in order to ensure accurate results. For this purpose, we use again the output of the polynomial interpolation. From this, additional constraints on the depth of the vertices can be defined as $z_i^{\prime} - \sigma \leq z_i \leq z_i^{\prime} + \sigma$. σ is set to a small value (e.g. 0.5 cm) depending on the object's deformations. Apart from these constraints, we need to define an objective function that is best suited to our particular problem. This objective function is defined as: summation over all relative depths, which is equal to a linear expression g in terms of the depth of the vertices (with coefficients -1, 0 or +1), depending on the direction of the edges. For example, using my conventional directions for a 9×9 mesh, we would have $\sum_{j=1}^{ne} rz_j = g = z_{73} - z_9$. As a result, the error e_z to be minimized will be:

$$e_z = \sum_{j=1}^{ne} rz_j - g \tag{3.10}$$

However, this error must be close to zero but strictly positive. Therefore, we need to specify $e_z \geq 0$. Finally the depth of all vertices can be estimated via the linear program expressed as:

$$\min_z e_z \quad \text{such that}$$

$$e_z = \sum_{j=1}^{n_e} rz_j - g, \;\; e_z \geq 0, \;\; z_i' - \sigma \leq z_i \leq z_i' + \sigma$$

$$\mathbf{Mx} = \begin{bmatrix} \mathbf{0} \\ \mathbf{p}_z \end{bmatrix}, \;\; z_i \geq 0, \;\; \forall i \in \left\{ 1, \;\; \cdots, \;\; n_v \right\} \tag{3.11}$$

where $\mathbf{M}$ is a $(n_{tr} + n_e + N) \times (n_e + n_v)$ matrix containing the coefficients of the linear system, $\mathbf{x}$ represents the vector comprising z_i and rz_j for all is and js and $\mathbf{p}_z$ indicates the set of all $p_{z,i}$s. This LP problem provides accurate estimates, as will be shown in the experimental results.

3.4.2 Step 2: Estimation of the *xy*-Coordinates of the Vertices

Assuming that $\mathbf{K}^{\circ}_{rgb}$ is the calibration matrix of the RGB camera, an optimization procedure is formulated to estimate the variables $\mathbf{q}^{\circ}_{v,i}$ ($\mathbf{q}^{\circ}_{v,i} = z_i \mathbf{q}_{v,i} = \begin{bmatrix} u^{\circ}_i & v^{\circ}_i \end{bmatrix}^T$) of vertex i. I call these variables *un-normalized image coordinates*. Such estimation is based on what I call *un-normalized projected lengths* and is performed by means of second-order cone programming (SOCP), consequently determining the full 3D positions of the vertices. This SOCP includes a linear objective function and a set of linear and conic constraints.

Unnormalized Projected Length: Let us represent $\mathbf{v}_1$ and $\mathbf{v}_2$ as $\mathbf{v}_1 = \begin{bmatrix} x_1 & y_1 & z_1 \end{bmatrix}^T$ and $\mathbf{v}_2 = \begin{bmatrix} x_2 & y_2 & z_2 \end{bmatrix}^T$, respectively . We can derive the difference between the corresponding unnormalized image points ($\mathbf{q}^{\circ}_{v,1} = \begin{bmatrix} z_1 u_1 & z_1 v_1 \end{bmatrix}^T$ and $\mathbf{q}^{\circ}_{v,2} = \begin{bmatrix} z_2 u_2 & z_2 v_2 \end{bmatrix}^T$) as follows: $z_1 u_1 - z_2 u_2 = f(x_1 - x_2), \;\; z_1 v_1 - z_2 v_2 = f(y_1 - y_2)$. By squaring and subsequently computing the sum of these two equations, we obtain this:

$$(z_1u_1 - z_2u_2)^2 + (z_1v_1 - z_2v_2)^2 = f^2\left[(x_1 - x_2)^2 + (y_1 - y_2)^2\right] \tag{3.12}$$

Note that the 3D length of an edge can be expressed as:

$$L^2 = (x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2 \tag{3.13}$$

For isometric deformations, the geodesics between any two points on the surface is constrained to a constant value. The Euclidean distance between these two points can be assumed to equal the corresponding geodesics when the edge connecting them is generally short-length and the deformations do not cause sharp creases along this edge. Therefore, let us assume that L does not change and can be pre-computed from the template. Note that z_1 and z_2 have been already determined. With these results Equation 3.13 can be rewritten as: $L^2 - (z_1 - z_2)^2 = (x_1 - x_2)^2 + (y_1 - y_2)^2$. Thus, the right-hand side of Equation 3.12 can be easily calculated with the equation above. We define as the *unnormalized projected length* the square root of the left-hand side of Equation 3.12

$$l = \sqrt{(z_1u_1 - z_2u_2)^2 + (z_1v_1 - z_2v_2)^2} \tag{3.14}$$

SOCP Optimization: Equation 3.14 introduces a quadratic constraint. Such a constraint may not be satisfied if folds between mesh vertices occur. To deal with this issue, we replace the above constraint by a variation that allows the vertices to move closer. So, it can be relaxed into a conic constraint as:

$$\sqrt{(u_{i,1}^{\circ} - u_{i,2}^{\circ})^2 + (v_{i,1}^{\circ} - v_{i,2}^{\circ})^2} \leq l \tag{3.15}$$

where $i \in \left\{ \begin{array}{ccc} 1, & \cdots, & n_e \end{array} \right\}$. The above conic constraint is applied to each edge of the mesh. According to Equation 3.1, the unnormalized image coordinates of feature point k (i.e. $\mathbf{q}_k^{\circ} = p_{z,k}\mathbf{q}_k = \left[\begin{array}{cc} u_{f,k}^{\circ} & v_{f,k}^{\circ} \end{array} \right]^T$) can be represented as

$$u_{f,k}^{\circ} = a_{k1}u_1^{\circ} + a_{k2}u_2^{\circ} + a_{k3}u_3^{\circ},$$

$$v_{f,k}^{\circ} = a_{k1}v_1^{\circ} + a_{k2}v_2^{\circ} + a_{k3}v_3^{\circ}, \tag{3.16}$$

where $k \in \left\{ \begin{array}{ccc} 1, & \cdots, & N \end{array} \right\}$. The linear equations above hold for all the feature points. In these equations the left-hand side represents the observed un-normalized image coordinates of the feature points, while the right-hand side represents the estimated coordinates. The cost function being minimized is the geometric distance between these two terms. However, in formulating our optimization as a SOCP, this error is not used as the objective function but as a conic constraint:

$$\left(\sum_{k=1}^{N} \left\| \left[\begin{array}{c} u_{f,k}^{\circ} \\ v_{f,k}^{\circ} \end{array} \right] - \sum_{j=1}^{3} a_{kj} \left[\begin{array}{c} u_j^{\circ} \\ v_j^{\circ} \end{array} \right]^{[k]} \right\|^2 \right)^{\frac{1}{2}} \leq \sigma_{uv} \tag{3.17}$$

Finally, the appropriate SOCP is formulated like this: $\min_{\mathbf{q}_v^{\circ}}$ œuv such that Equations 3.15 and 3.17 are satisfied. When applied to a number of different mesh deformations of a generic isometric surface, the approach detailed in this section results in the training data set required to reconstruct other isometric surfaces by the use of only a normal camera and from a single view, as discussed in the previous section.

3.5 Experiments and Results

3.5.1 Synthetic Data

With synthetic data that exactly simulates and conforms to various deformations of a 9×9 mesh, I have evaluated both reconstruction schemes proposed, in order to validate their efficiency. The evaluation comprised a number of experiments, conducted with a set of feature points ($N = 60$) well distributed over the mesh triangles. From the planar template mesh, the barycentric coordinates can be computed - see Figure 3.3. The virtual RGB camera model is defined such that the focal length is $f = 268$ pixels. With this model, point correspondences across the simulated deformations were projected onto the virtual RGB image plane, assuming that the simulated mesh is placed 50 *cm* in front of the camera (along the optical axis). To perform the quantitative evaluation it is necessary to define some numerical metrics as follows:

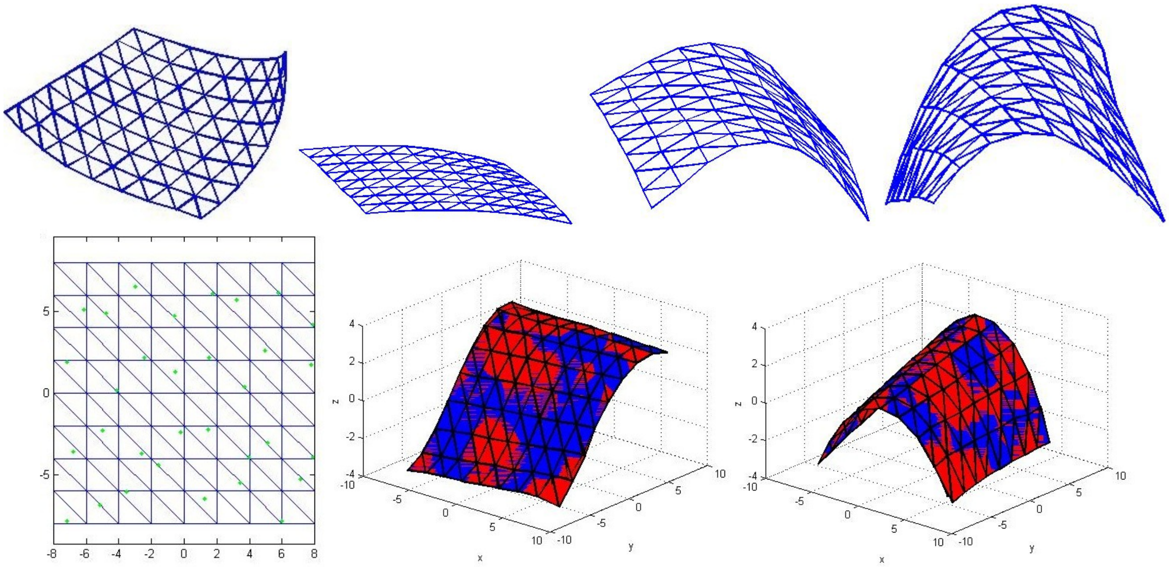

Figure 3.3: Top - Some simulated deformations. Bottom - Left: A 9×9 template mesh with feature points - radius = 20 cm. Right: Metric coordinates in cm - overlap between the ground-truth shape (blue) and the estimated shape (red).

- To evaluate the results from mesh depth recovery, obtained by linear programming the following criterion is adopted:

  $DepthAccuracy = \frac{1}{n_v} \sum_{i=1}^{n_v} \left[||z_{v,i} - \hat{z_{v,i}}||^2 \,/\, ||\hat{z_{v,i}}||^2 \right]$

  Mesh depth estimates are strongly affected by errors mainly due to the errors on the depth estimates of the feature points - see Figure 3.4.

- Point reconstruction error (PRE): the normalized Euclidean distance between the observed ($\hat{\mathbf{p}}_i$) and estimated ($\mathbf{p}_i$) feature points:

  $PRE = \frac{1}{N} \sum_{i=1}^{N} \left[||\mathbf{p}_i - \hat{\mathbf{p}}_i||^2 \,/\, ||\hat{\mathbf{p}}_i||^2 \right]$

- Mesh reconstruction error (MRE): The normalized Euclidean distance between the observed ($\hat{\mathbf{v}}_i$) and estimated ($\mathbf{v}_i$) 3D vertices of the mesh, computed as:

  $MRE = \frac{1}{n_v} \sum_{i=1}^{n_v} \left[||\mathbf{v}_i - \hat{\mathbf{v}}_i||^2 \,/\, ||\hat{\mathbf{v}}_i||^2 \right]$

- The re-projection error of the feature points is also another measure of precision:

  $ReprErr = \frac{1}{N} \sum_{i=1}^{N} \left[||q_i - \hat{q}_i||^2 \,/\, ||\hat{q}_i||^2 \right]$

- The standard deviation of the errors on the estimates of 3D positions of the mesh vertices: the standard deviation of the global error in each coordinate of the mesh vertices estimated with the monocular optimization algorithm (calculated separately for each coordinate).

Note that all quantitative results represent an average obtained from five deformations randomly selected. By performing 500 trials for each deformation, each average value was acquired from 2500 trials.

Experiments on D-RGB Based Reconstruction: I obtained results in a set of experiments where Gaussian random noise with five different standard deviation values added to the depth of the synthetic feature points. Noise levels with standard deviations greater than 0.3 cm prevent the LP from giving good results, as shown in Figure 3.4. Since image points

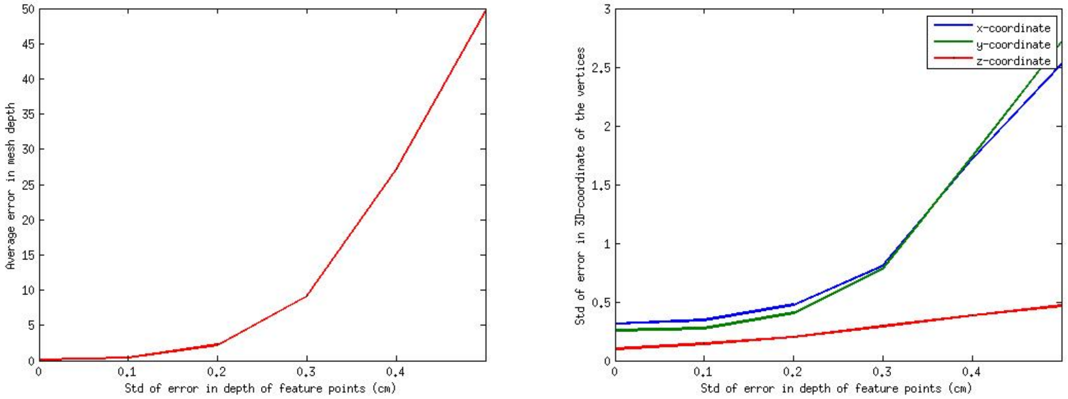

Figure 3.4: D-RGB based reconstruction-based - Left: The error on the depth estimates of the mesh vertices, computed via LP. Right: The std of the global error on the estimates of the positions of the mesh vertices.

are also used in the 3D reconstruction, the effect of the noise in the image points was evaluated. To do so, Gaussian noise was also added to the image points (with standard deviation in increments of 0.5 pixels). Figure 3.6 shows how the reconstruction accuracy behaves as a function of noise level. In the left-hand plot, a zero-mean Gaussian noise with $0.1 - cm$ std in the depth estimates of the feature points was also considered in all the relevant tests. In the right-hand plot, on the other hand, a zero-mean Gaussian noise with 1-pixel standard deviation in image points was also considered in all the relevant tests. Two of the recovered deformations and their equivalent ground-truth are illustrated in Figure 3.3.

Experiments on Monocular Reconstruction: To perform such experiments a deformation model has to be estimated. This model was directly obtained by applying PCA to a comprehensive set of synthetic mesh deformations. In these experiments Gaussian noise was also added to the image points (with standard deviation in increments of 0.5 pixels). Figure 3.7 shows how the reconstruction accuracy behaves towards noise.

Comparative Evaluation: In the literature there are several approaches for 3D reconstruction of deformable surfaces. To compare against the approaches described in this chapter I chose the approach presented in [32]. The main reason for selecting this work is because the authors have used both the linear deformation model and the mesh representation and this enables us to make reliable comparisons. They also propose a SOCP problem for the reconstruction and their approach is known to be robust and efficient, in which a linear local deformation model is used to combine local patches into a global surface. I obtain results from this approach and the proposed reconstruction schemes in the absence of noise. The linear deformation model for the approach being compared and my monocular reconstruction was computed from the results of the D-RGB based reconstruction of the synthetic mesh deformations. Two different cases have been examined: 1- simple deformations with small, moderate creases 2- complex deformations with large, sharp creases. Accordingly, the comparative results are divided in two cases, as shown in Figure 3.5. The charts reveal that the monocular reconstruction outperforms the other two approaches in the case of simple deformations but its performance declines significantly in the case of complex deformations, while the D-RGB based reconstruction maintains satisfactorily stable performance under different situations (i.e. the results do not vary dramatically).

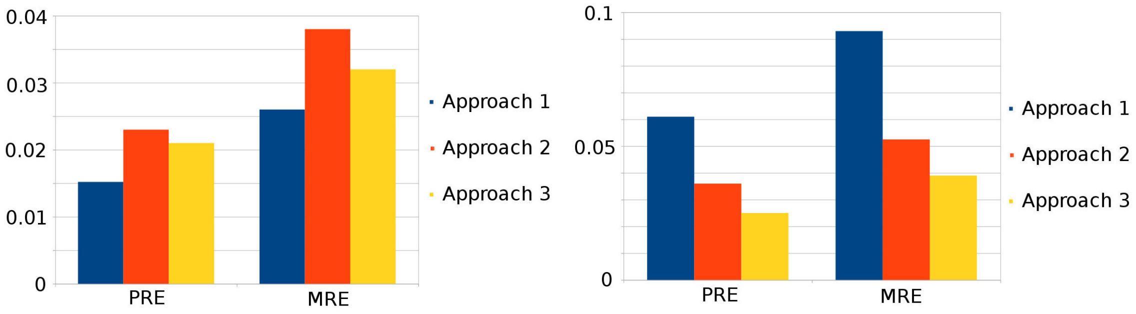

Figure 3.5: Left-hand chart: the case of simple deformations; Right-hand chart: the case of complex deformations. Approaches 1, 2 and 3 refer to my monocular reconstruction, the approach presented in [32] and the D-RGB based reconstruction, respectively.

3.5.2 Real Data

For qualitative assessment, the reconstruction schemes have been tested with real data. A camera setup made up of a high-quality ToF camera and a high-resolution RGB camera

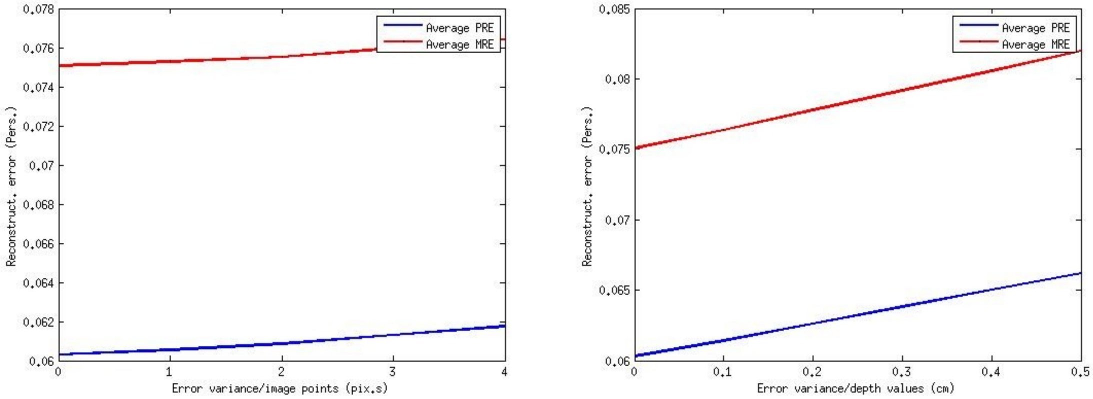

Figure 3.6: D-RGB based reconstruction - Left: Average PRE and average MRE with respect to the increasing noise in image points. Right: Average PRE and average MRE with respect to the increasing noise in depth data.

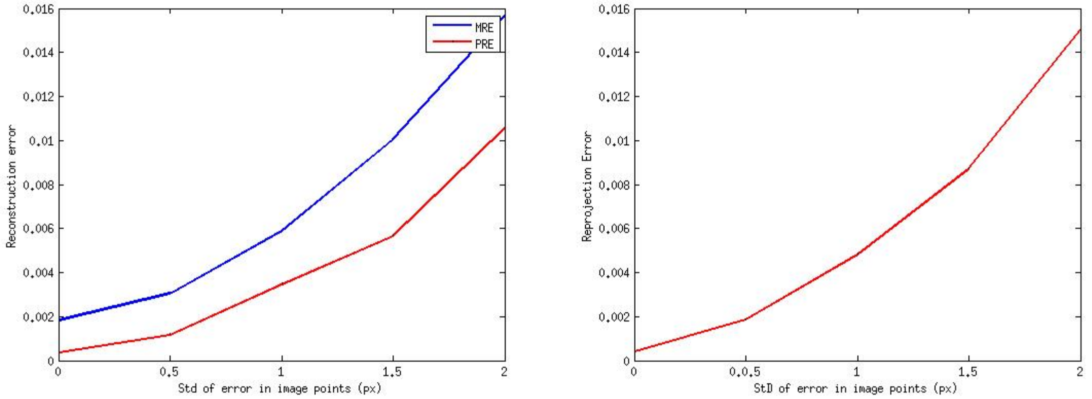

Figure 3.7: Monocular reconstruction - Metrics as a function of the noise in image points. Left: Average PRE and average MRE. Right: Average reprojection error.

was prepared for D-RGB based reconstruction. The two cameras were calibrated both internally and externally. In the experiments I used a piece of cardboard flexible enough to allow creating as many different deformations as possible so that the deformation model learned from the reconstruction results could be generalized to other surfaces of different material. The camera setup was located 60 cm in front of the surface being reconstructed, guaranteeing that the FOV of the ToF camera was completely covered by the RGB camera. A regular 9×9 mesh was again used to represent the surface, with the positions of the feature points available in relation to the positions of the vertices on the planar template.

Such positioning data enabled the calculation of the barycentric coordinates for the feature points. Correspondence of these points across the image sequence was established with respect to the template. The depth of the points, given by the ToF image, was registered with respect to the RGB image. The rest of the implementation was just the same as in the experiments with synthetic data. After having applied the D-RGB based reconstruction to multiple deformations of the cardboard, the training data necessary to estimate the deformation model was acquired, as shown in Figure 3.8. In the case of monocular reconstruction, this deformation model was then employed to reconstruct the same mesh overlaid across such isometric surfaces as those given in Figure 3.9, by using only the RGB camera. The qualitative results have shown that this reconstruction scheme yields good results (although no quantitative assessment was possible because of lack of the ground truth). It is worth mentioning that the left and right surfaces in Figure 3.9 resemble the cardboard in Figure 3.8 in terms of flexibility in particular, whereas the middle surface was made with a different material. However I reached the conclusion that while the results for the left and right surfaces appeared slightly better than those for the middle surface, a readily deformable cardboard is a proper choice for deriving the linear deformation model.

3.6 Conclusions

We dealt with reconstruction of isometric surfaces. To perform such monocular reconstruction, an algorithm based on the linear deformation model and consisting of a non-linear least squares optimization was proposed. To find the proper deformation model, prior training data should be used. I therefore provided this prior data by proposing a novel approach for the reconstruction of a typical surface so that the computed deformation model can be also extended to other isometric surfaces. This approach was founded on a range camera along with a conventional camera and its goal is to estimate the 3D positions of the mesh vertices from the depth of the feature points. By applying this approach to multiple mesh

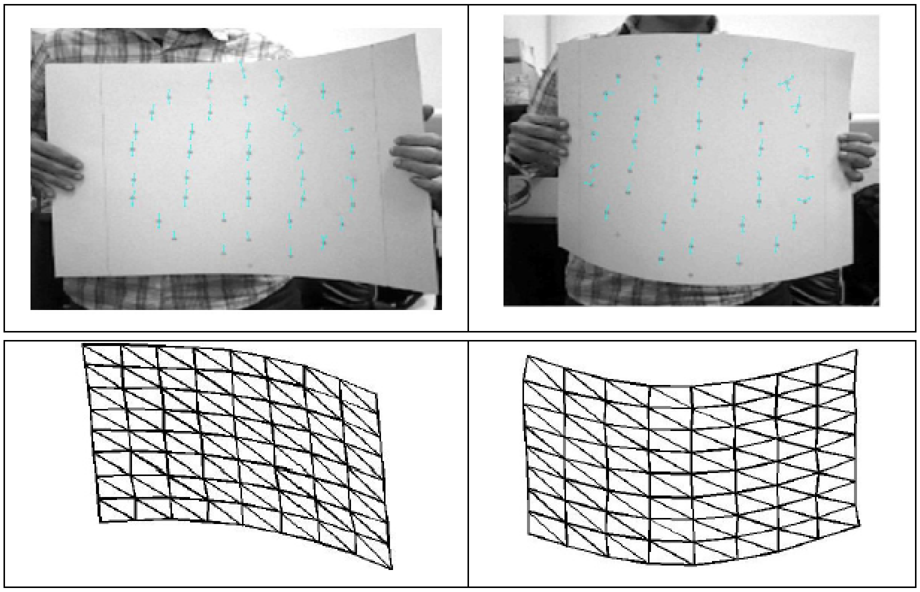

Figure 3.8: Real deformations; A 20 × 20-cm square was selected from the intermediate part of the cardboard and then reconstructed.

Figure 3.9: Isometric surfaces: Real deformations. Courtesy of [33].

deformations I acquired the training data required. Experimental results showed that both the proposed reconstruction schemes are efficient and result in accurate reconstructions.

CHAPTER 4

DEFORMATION CONSTRAINTS

4.1 Introduction

This chapter discusses the second task in detail and addresses two reconstruction schemes, each in one subtask. Both schemes are developed on a similar basis which is the joint use of deformation constraints based on the upper-bound model and on the differential properties.

The first scheme is assigned to monocular reconstruction of surfaces that deform isometrically, from points tracked on a single image. To tackle this problem, a flat 3D shape of the surface and its image are used as template. Such deformations involve certain geometric constraints. To reconstruct them, these constraints should be properly exploited. Therefore we propose an algebraic formula that aims at the joint expression of the geometric constraints formerly introduced in this context, namely based on the differential model and on the upper-bound model. This expression is the result of integrating these two types of constraints and leads to the intended reconstructions, even when the surface is not strictly isometric. The template shape is used to set the parameters of this expression, which is then used along with the projection equations in order to define a semi-definite programming problem to enable the estimation of 3D positions of the points throughout the surface. However, for implementation purposes, this optimization is applied separately to patches which, together, make up the whole surface. The experimental results show that the proposed approach improves the results from other methods in terms of accuracy.

We also study monocular reconstruction of extensible surfaces undergoing conformal deformation and propose a reconstruction scheme for this sort of surfaces. Given a 3D template, its image and a set of point correspondences from a specific image of the deformed surface, the objective is to determine the 3D positions of the points as well as the stretching factor at each point. To perform such reconstruction, I define an optimization procedure that makes use of the re-projection error, so-called *upper-bound* model, constraints associated with conformal deformation, and those resulting from the assumption that the motion amplitude of boundary points is restricted. These points lie on the surface border which varies by the camera viewpoint. The upper-bound model is used in a variational form where the bound on the Euclidean distance of point pairs decreases in a sequence of optimizations, from which the best reconstruction is selected. My experimental study revealed that this approach achieves accurate results.

Isometric Surface Reconstruction

4.2 Model and Approach

In this subtask, I address the reconstruction of inextensible deformable surfaces, also known as isometric, from a single image. The prior data consists of a flat 3D shape of the surface and its image. These are used as a template. Given a set of 2D point/feature correspondences between an image and the template image, the objective is to estimate their 3D positions. My algorithm uses a specific representation of the differential models that allows the reconstruction to be performed with reduced computational cost. Furthermore, I combine the upper-bound model with the differential model in order to obtain improved reconstruction accuracy, when the surface is not perfectly isometric. Although I also employ a mapping for the differential model, its parameters will be estimated after the computation of the 3D positions, whereas in the case of the approaches [56, 57], the mapping parameters are estimated first. That fact enables us to obtain more precise estimates of the 3D positions of the feature points. In addition, it is also possible to interpolate the 3D coordinates of other points through the mapping, which results in a smooth reconstruction of the surface. Those estimates might not be as accurate as those obtained by means of the approaches attempting to recover the shape of a deformed surface as a whole (independently of any points lying on that surface). However, the problem addressed here refers to point-wise reconstruction and, in what concerns that type of reconstruction, the proposed approach performs well.

The approach proposed here assumes that an image of the deformed surface is available and that feature/point correspondences between this image and the template image are obtained. The combination of the constraints resulting from the differential and from the upper-bound models is represented analytically by quadratic equations. The template shape is used to estimate the parameters of the deformation constraints. The re-projection error is formulated as a quadratic error. The deformation constraints and the reprojection error make up the core formulation of my algorithm. The differential model requires a 3D warp whose role is to map the template image onto the deformed 3D surface. In my work, this 3D warp is described parameterically by the use of Free-Form Deformations (FFD). The optimization is formulated as a Semi-Definite Program. My approach is built on a patch-based strategy, in the sense that the surface is divided into several patches that, together, cover the entire surface, and the method is applied to each one of them separately. This is done to circumvent limitations that occur during the computation. As a matter of fact, surface patching allows to reduce the implementation complexity (due to the scale of the problem) so that the computational cost can be handled efficiently. It also facilitates the use of FFD, as detailed in the next section. In addition, it provides a convenient way to identify subregions with a uniform point distribution in case the overall surface is represented by non-uniformly distributed points. As a result of recovering the 3D positions of the feature points, the parameters of the mapping can be estimated, thereby allowing the estimation of the 3D positions of points other than those corresponding to the point correspondences. This leads to a 3D dense reconstruction of the surface.

Organization of this subtask: First, I discuss the constraints in detail, namely, the deformations constraints and the reprojection error. The next section presents the SDP optimization formulation. Then, the experiments and their results are described, followed by conclusions at the end.

4.3 Proposed Approach: Constraints

I propose a reconstruction method for surfaces whose deformation complies with the specified deformations constraints. Those constraints are derived from the combination of both the upper-bound model and the differential formulation of the deformation.

4.3.1 Deformation Constraints: Differential Model

Consider a surface $\mathbf{S} \subset \mathbb{R}^3$ with the following parametric representation:

$$\mathbf{p}\left(u, v\right) = \left(x\left(u, v\right), y\left(u, v\right), z\left(u, v\right)\right) \tag{4.1}$$

in some domain in $\mathbb{R}^2$. The coefficients of the First Fundamental Form (FFF) of the surface in $\mathbb{R}^3$ can then be arranged in a symmetric matrix as:

$$\mathbf{M} = \begin{bmatrix} E & F \\ F & G \end{bmatrix} \tag{4.2}$$

Theorem: A surface is isometric if and only if the coefficients of FFF are the same, i.e., $\mathbf{M} = \mathbf{M}^*$. Two surfaces are said to be isometric if there exists an isometry between them. All possible deformations of an isometric surface have the same Gaussian curvature at corresponding pairs of points (since Gaussian curvature depends only on the FFF). The 3D template enables the calculation of the FFF of the surface at every point. Hence, $\mathbf{M}$ is known in advance. Let us assume that $\mathbf{M}$ represents the FFF coefficients for the 3D template, and that $\mathbf{M}^{'}$ represents the FFF coefficients for the deformed surface $\mathbf{S}$ to be reconstructed. Assuming the following equation

$$x_1 = \frac{\delta x}{\delta u}, \ \ x_2 = \frac{\delta x}{\delta v}, \ \ y_1 = \frac{\delta y}{\delta u}, \ \ y_2 = \frac{\delta y}{\delta v}, \ \ z_1 = \frac{\delta z}{\delta u}, \ \ z_2 = \frac{\delta z}{\delta v} \tag{4.3}$$

for the surface **S**, then for point *i* we will have:

$$E_i^{'} = x_{i,1}^2 + y_{i,1}^2 + z_{i,1}^2 = E$$

$$G_i^{'} = x_{i,2}^2 + y_{i,2}^2 + z_{i,2}^2 = G$$

$$F_i^{'} = x_{i,1}.x_{i,2} + y_{i,1}.y_{i,2} + z_{i,1}.z_{i,2} = F \tag{4.4}$$

As a result, the expressions for $E_i^{'}$ and $G_i^{'}$ are quadratic with respect to the partial derivatives. We also exploit a quadratic equation for the $F_i^{'}$, implying that $F_i^{'}$ can be replaced by the expression for $H_i^{'}$, which is also quadratic, similarly to $E_i^{'}$ and $G_i^{'}$. This expression can be easily derived by representing each cross-term of partial derivatives with respect to the squared sum of the two derivatives involved.

$$H_i^{'} = (x_{i,1} + x_{i,2})^2 + (y_{i,1} + y_{i,2})^2 + (z_{i,1} + z_{i,2})^2 = E_i + G_i + 2.F_i = H_i \tag{4.5}$$

Representation of $E_i^{'}$, $G_i^{'}$ and $H_i^{'}$ in terms of the point set **p**:

As mentioned earlier, the coordinates of the 3D points can be computed as the output of a 3D warp. The use of the 3D warp allows the derivation of the differential model.

3D warping can be also regarded as a method of 3D reconstruction, in which a template image is mapped onto the 3D object, a surface for instance [58, 59]. Regular surface reconstruction procedures involve a minimization over all possible variations of the surface in space. However, the application of a 3D warp is a common approach to reducing the optimization search space so that the search can be performed efficiently. This reduction is achieved by using a parametric representation of the surface (e.g. by means of a 3D warp). It is possible to model a 3D warp using a triangular mesh, which is a usual choice in deformable surface reconstruction [32, 40, 60, 61]. Frequent 3D warps found in the literature are FFD warps. My approach also uses a FFD-based 3D warp, to model the mapping from the template image onto the 3D surface $\mathbf{S}$. With this mapping, the surface points $\mathbf{p}$ corresponding to the points $\mathbf{q} = \begin{pmatrix} u, & v \end{pmatrix}$ on the template image are given by:

$$\mathbf{p}(\mathbf{q}) = \sum_{i=0}^{n} \sum_{j=0}^{m} \mathbf{b}_{i,j} N_{i,k}(u) M_{j,l}(v) \tag{4.6}$$

where $N_{i,k}(u)$ and $M_{j,l}(v)$ are B-spline basis functions of degree k and l, respectively. $\mathbf{b}_{i,j}$ act as a set of $n_b = (n+1) \times (m+1)$ control points which, as in Bézier surfaces, is usually referred to as the control net. For a point on the template image, the product $N_{i,k}(u)M_{j,l}(v)$ will be constant, which could be regarded as a coefficient. As a result of computing this product for each ij, the corresponding surface point can be represented as a linear combination of all $\mathbf{b}_{i,j}$, according to Equation 4.6. Therefore, the set resulting from putting together the linear expressions for n_p points is rewritten as a linear system of equations: $\mathbf{p} = \mathbf{A}_c\mathbf{b}$ where $\mathbf{p}$ is the set of all surface points, $\mathbf{A}_c$ denotes the $3n_p \times 3n_b$ collocation matrix obtained from all products $N_{i,k}M_{j,l}$ and $\mathbf{b}$ is the vector formed by concatenating all $\mathbf{b}_{i,j}$. A single solution is obtained when the number of linearly independent equations is equal to that of the unknowns. Each point removes a degree of freedom, thus adding an independent equation to the linear system. Therefore, we need as many as n_b points in order for this system to have a single solution, which can be determined by least squares minimization provided that **A** has full column rank (as is usually the case with our particular problem). So, the solution of the linear system can be obtained from:

$$\mathbf{b} = \left(\mathbf{A}_c^T \mathbf{A}_c\right)^{-1} \mathbf{A}_c^T \mathbf{p} \tag{4.7}$$

This solution probably will not be valid for other points on the surface and there might be errors. Note that an overdetermined linear system has no exact solution. In addition, a square full-rank linear system admits of a strictly exact solution that only fits the measurements included in the linear system. One possibility of obtaining a more general solution is to use an undetermined system of equations, with additional degrees of freedom. However it should be mentioned that, depending on the surface to be reconstructed, the number of control points is usually limited in order to avoid complex control nets. For example, a typical control net contains 5×5 control points. On the other hand, a smooth reconstruction of the surface requires that numerous feature points be distributed across the surface. However, by dividing the entire surface into small square-like patches as an alternative technique, we can balance the limited set of control points against the large number of feature points.

Using this method, the reconstruction is carried out as follows: **a.** $n_p < n_b$ feature points exist on each patch; **b.** a distinct linear system $\mathbf{p} = \mathbf{A}_c \mathbf{b}$ is defined for each patch; **c.** Each patch is reconstructed separately. In this case, **b** will be approximated by

$$\left(\mathbf{A}_{cb}^T \mathbf{A}_{cb}\right)^{-1} \mathbf{A}_{cb}^T \mathbf{A}_b \mathbf{p} \tag{4.8}$$

where $\mathbf{A}_{cb} = \mathbf{A}_c^T \mathbf{A}_c + \mathbf{A}_b$ and $\mathbf{A}_b$ denotes the bending matrix which, as defined in [56], is a $n_b \times n_b$ symmetric, positive semi-definite matrix which can be computed from the second derivatives of the B-spline basis functions. For simplicity's sake, we substitute $\mathbf{A}^\circ$

for $\left(\mathbf{A}_{cb}^T\mathbf{A}_{cb}\right)^{-1}\mathbf{A}_{cb}^T\mathbf{A}_b$, leading to $\mathbf{b} = \mathbf{A}^{\circ}\mathbf{p}$.

The partial derivatives associated with the coefficients $E_i^{'}$, $G_i^{'}$ and $H_i^{'}$ can be obtained using Equation 4.6, thereby expressing them as a linear combination of the control points **b**. In addition, we have $\mathbf{b} = \mathbf{A}^{\circ}.\mathbf{p}$, so these partial derivatives can be also represented linearly in terms of **p**. Our goal is to estimate vector **p**, containing all the points. Therefore the coefficients $E_i^{'}$, $G_i^{'}$ and $H_i^{'}$ can be expressed in terms of **p**. Consequently, for point *i*, we have:

$$E_i^{'} = \mathbf{p}^T\mathbf{W}_{E,i}\mathbf{p},\ G_i^{'} = \mathbf{p}^T\mathbf{W}_{G,i}\mathbf{p},\ H_i^{'} = \mathbf{p}^T\mathbf{W}_{H,i}\mathbf{p} \tag{4.9}$$

where $\mathbf{W}_{E,i}$, $\mathbf{W}_{F,i}$ and $\mathbf{W}_{G,i}$ are sparse constant matrices, associated with point *i*.

4.3.2 Deformation Constraints: Upper-Bound Model

This model is basically applied to the reconstruction of isometric surfaces and formulated taking into account that the inextensibility constraint $\left\|\mathbf{p}_j - \mathbf{p}_k\right\| \leq d_{j,k}$ must be satisfied by any pair of points $(\mathbf{p}_j, \mathbf{p}_k)$ lying on the surface. Note that d_{jk} is the geodesic distance, which is equal to the Euclidean distance on the planar 3D template. In practice, the selected point pairs are often reduced to take into account only those connecting close points. Such points can be found by limiting the distance between every two points on the template to a given threshold. This constraint is usually known as the the upper-bound model. A second constraint has been introduced in the literature along with this model, in order to allow for the reconstructions. This constraint results from a heuristic and imposes the maximization of the depth of the 3D points to be reconstructed. In this case, the reconstruction approach combines the upper-bound model and the second constraint, also with the data re-projection error. The optimization is formulated as a second-order-cone program (SOCP) [32].

The upper-bound model $\left\|\mathbf{p}_j - \mathbf{p}_k\right\| \leq d_{j,k}$ can be rewritten as:

$$\mathbf{p}^T \mathbf{C}_{j,k} \mathbf{p} \leq d_{j,k}^2 \tag{4.10}$$

where $\mathbf{C}_{j,k}$ is a sparse constant matrix, associated with the point pair j,k.

4.3.3 Reprojection Error

For point i, the re-projection error is computed as $eu_i = \left(f.p_{x,i} - u_i^{'}.p_{z,i}\right)^2$ and $ev_i = \left(f.p_{y,i} - v_i^{'}.p_{z,i}\right)^2$, which will be then represented as follows:

$$eu_i = \mathbf{p}^T \mathbf{R}_{u,i} \mathbf{p} = 0, \;\; ev_i = \mathbf{p}^T \mathbf{R}_{v,i} \mathbf{p} = 0 \tag{4.11}$$

where $\mathbf{R}_{u,i}$, $\mathbf{R}_{v,i}$ are sparse constant matrices, associated with the point i.

4.3.4 Proposed Approach: SDP Optimization Scheme

As described earlier, we aim at estimating the 3D positions of the points $\mathbf{p}$. To do so, we develop an optimization procedure based on SDP, which has proved to be efficient when the problem under consideration is subject to quadratic constraints. Such constraints basically render the whole problem intractable (non-convex). Most interior-point methods for linear programming have been generalized to semidefinite programs. As in linear programming, these methods have polynomial worst-case complexity, and perform very well in practice at an acceptable computational cost. The constraints in Equations 4.9 and 4.11 are equivalent to the following matrix traces:

$$Tr\left(\mathbf{W}_{E,i}\mathbf{p}\mathbf{p}^T\right) = E_i, \; Tr\left(\mathbf{W}_{G,i}\mathbf{p}\mathbf{p}^T\right) = G_i, \; Tr\left(\mathbf{W}_{H,i}\mathbf{p}\mathbf{p}^T\right) = H_i$$

$$Tr\left(\mathbf{R}_{u,i}\mathbf{p}\mathbf{p}^T\right) = 0,\ Tr\left(\mathbf{R}_{v,i}\mathbf{p}\mathbf{p}^T\right) = 0 \tag{4.12}$$

This is a homogeneous system of quadratic equations on the unknowns $\mathbf{p}$. However it can be relaxed to a convex program. The quadratic equality constraints are non-convex. To make the problem convex, an auxiliary square matrix $\mathbf{D} = \mathbf{p}\mathbf{p}^T$ is introduced. A constraint that restricts the matrix $\mathbf{D}$ to a positive semi-definite matrix ($\mathbf{D} \geq 0$) should then be added by the use of a SDP. This procedure is commonly applied to many problems [62]. As a result, the above system becomes a constrained optimization problem linear in $\mathbf{D}$, and also subject to a rank constraint in $\mathbf{D}$. The rank constraint is non-convex too. To deal with the rank-1 condition of $\mathbf{D}$, a common practice is to simply drop it. However, my preliminary experiments demonstrated that an optimization based on such an approach yields matrices D with small eigenvalues. It is also obvious that minimizing $rank\left(\mathbf{D}\right)$ is a weak relaxation and converges to trivial solutions. Instead, maximizing $rank\left(\mathbf{D}\right)$ is a preferred choice in such cases and leads to the desired convex program. To approximate rank, we use its nuclear norm, i.e. $\|\mathbf{D}\|_*$, which in my case reduces to trace since $\mathbf{D}$ is positive semi-definite. More precisely stated, maximizing the nuclear norm of $\mathbf{D}$ can be found to be equivalent to the depth maximization introduced in the subsection describing upper-bound model.

It is quite possible that a real deformation is not strictly isometric and slightly deviates from the isometry constraints. Such deformations can be called *near-isometric*. In that case, we need to account for the deviations. For this purpose, a near-isometric deformation is approximated by a more flexible one (in some particular cases, a conformal one) by adding the condition that stretching of the surface at each point can occur, implying that E_i, G_i, and H_i at point i are limited to a range of values corresponding to a slight stretching of the template. Assuming that $E_{i,min} = E_i - \lambda_{E,i}$, $G_{i,min} = G_i - \lambda_{G,i}$, $H_{i,min} = H_i - \lambda_{F,i}$, $E_{i,max} = E_i + \lambda_{E,i}$, $G_{i,max} = G_i + \lambda_{G,i}$ and $F_{i,max} = H_i + \lambda_{F,i}$, the optimization problem can be formulated as follows:

$$\max_{\mathbf{D}}\ Tr\left(\mathbf{D}\right),\ \text{subject to}$$

$$E_{i,min} \leq Tr\left(\mathbf{W}_{E,i}\mathbf{D}\right) \leq E_{i,max},$$

$$G_{i,min} \leq Tr\left(\mathbf{W}_{G,i}\mathbf{D}\right) \leq G_{i,max},$$

$$H_{i,min} \leq Tr\left(\mathbf{W}_{F,i}\mathbf{D}\right) \leq H_{i,max},$$

$$Tr\left(\mathbf{R}_{u,i}\mathbf{D}\right) = 0,\ Tr\left(\mathbf{R}_{v,i}\mathbf{D}\right) = 0,$$

$$\mathbf{D} \geq 0,\ Tr\left(C_k\mathbf{D}\right) \leq d_k^2 \tag{4.13}$$

where $i \in (1,\ldots,n_p)$ and $k \in (1,\ldots,l)$. l is the number of edges linking two close points. $\lambda_{E,i}$, $\lambda_{G,i}$ and $\lambda_{F,i}$ are set empirically. Vector **p** can be determined up to a sign ambiguity using SVD to find a rank-1 factorization of **D**. As for the missing points, we could fill them in by using estimates obtained with Equation 4.6.

When the deformation is completely isometric, the following optimization leads to improved results:

$$\max_{\mathbf{D}}\ Tr\left(\mathbf{D}\right),\ \text{subject to}$$

$$Tr\left(\mathbf{W}_{E,i}\mathbf{D}\right) = E_i,\ Tr\left(\mathbf{W}_{G,i}\mathbf{D}\right) = G_i,\ Tr\left(\mathbf{W}_{H,i}\mathbf{D}\right) = H_i$$

$$Tr\left(\mathbf{R}_{u,i}\mathbf{D}\right) = 0,\ Tr\left(\mathbf{R}_{v,i}\mathbf{D}\right) = 0,$$

$$\mathbf{D} \geq 0, \tag{4.14}$$

The first SDP differs from the above SDP in that it is mainly implemented using the differential isometric constraints rather than the inextensibility constraints. The SOCP optimization described in the subsection assigned to the upper-bound model. is a formulation of the maximum-depth heuristic. Without the differential near-isometric constraints, the SDP optimization given in Equation 4.13 would be equivalent to this SOCP in terms of the constraints considered, and with no loss of generality it can be said that the SDP that results from excluding the differential constraints is an alternative way of implementing the maximum-depth heuristic. As shown by my preliminary experiments, such SDP obtains essentially the same results as the SOCP.

Figure 4.1: Pictures of the paper-sheet, T-shirt and the cardboard, respectively. The paper-sheet is modeled using 5 patches, T-shirt using 4 patches and the cardboard using 3 patches (courtesy of [33])

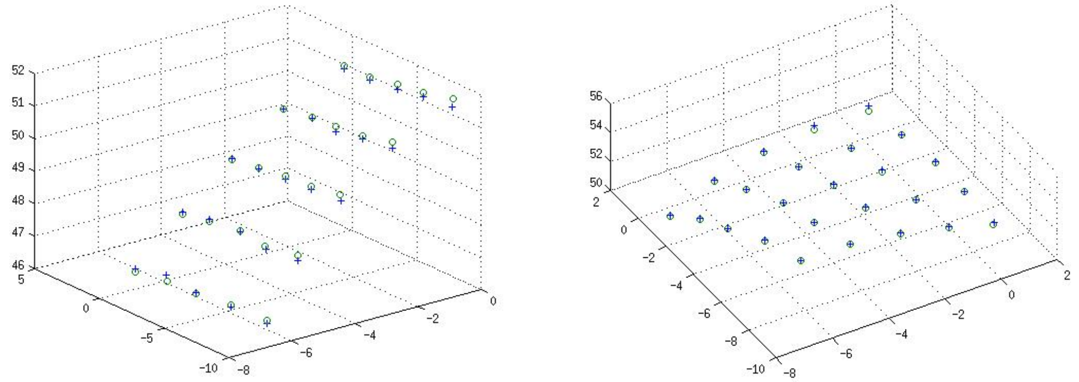

Figure 4.2: The points marked with '+' and those marked with 'o' illustrate the estimated and the ground-truth points on a single patch of the cardboard, respectively. The ground-truth was acquired using a motion capture device.

4.4 Experimental Results

In this section, I characterize the performance of my approach for different deformations of isometric surfaces. In all the experiments, only the optimization in Equation 4.13 has been considered and analyzed.

4.4.1 Experiments with Synthetic Data

To evaluate my approach quantitatively, a set of experiments with synthetic data was performed. However, the 3D data used in these experiments was not obtained through simulation. Rather, it was acquired from real surfaces expected to undergo isometric deformation. The reason for using such data is that I aimed at assessing my approach with real deformations, including near-isometry. This data includes the 3D positions of several non-rigid points located on surfaces such as paper-sheets, T-shirts and card-boards, which are likely to undergo inextensible deformations, and can be seen as proper examples of isometric surfaces. In fact, a set of n_c point correspondences has been acquired by a motion capturing device or a Kinect camera from these surfaces. There is only a subtle difference relative

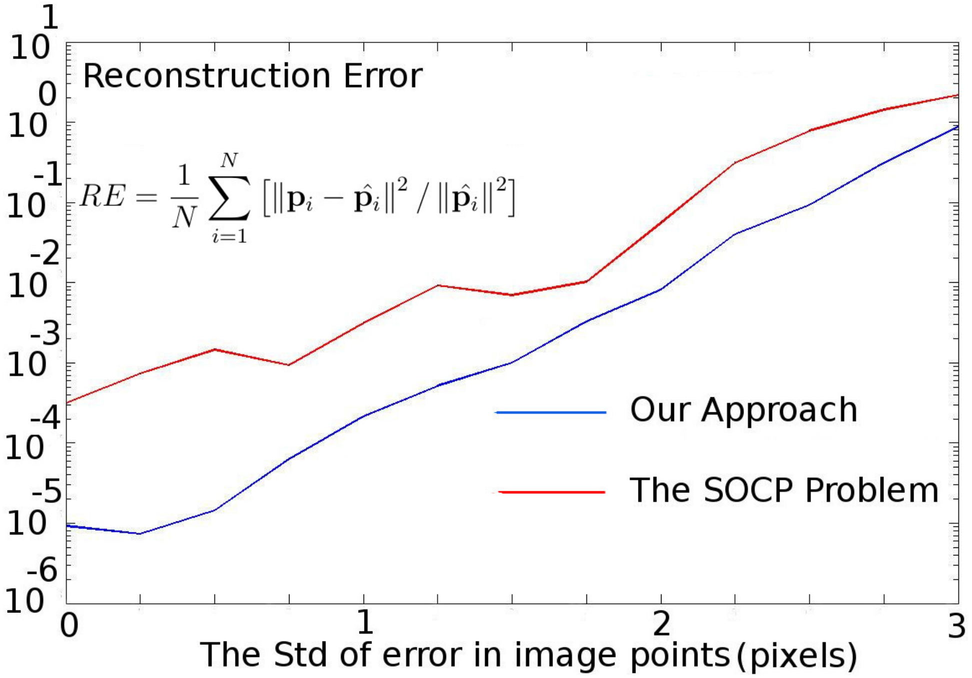

Figure 4.3: Reconstruction error for increasing levels of noise computed for my approach and for the SOCP optimization described in the subsection detailing the upper-bound model

Approach	Reconstruction Error
My Approach	9.2752e-06
Approach in [56]	5.3346e-05
Approach in [32]	7.3317e-05

Table 4.1: A comparison between three approaches: my approach, the approach of [56] and the approach of [32].

to my experiments with real data in that, in this case, a virtual camera was considered to obtain the 2D projections. Therefore, the synthetic image points were obtained by using a virtual camera. Photos of the surfaces are shown in Figure 4.1. The accuracy of the results is reported in terms of reconstruction error which is the normalized Euclidean distance between the observed ($\hat{\mathbf{p}}_i$) and the estimated ($\mathbf{p}_i$) world points:

$$RE = \frac{1}{N}\sum_{i=1}^{N}\left[\|\mathbf{p}_i - \hat{\mathbf{p}}_i\|^2 / \|\hat{\mathbf{p}}_i\|^2\right] \tag{4.15}$$

It is assumed that an average of 30 points lies on each patch with a control net of 6×6,

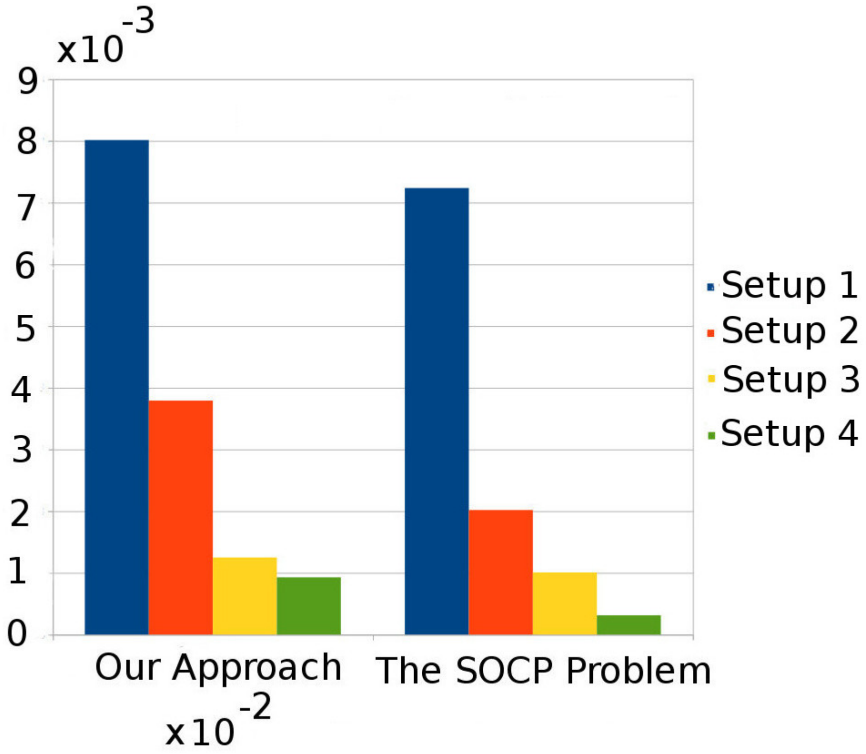

Figure 4.4: Reconstruction error for different camera setups. In the case of my approach $\times 10^{-2}$ indicates that the errors correspond to the associated columns scaled by $\times 10^{-2}$.

with a varying number of patches, depending on the surface to be reconstructed. The reconstruction is performed under the assumption that there is a relatively uniform density distribution of the feature points on each patch. In addition to creating a balance between feature and control points, surface patching is used for two major purposes: first, matrix $\mathbf{D}_i$ will be of moderate size (which would otherwise be extremely large), thus limiting the computational cost. Second, if in some case the overall surface has a non-uniform point distribution, surface patching helps to identify subregions with a close-to-uniform point distribution, and subsequently consider them as patches to be reconstructed separately. These patches are assumed to be non-overlapping and each one of them is treated like a whole surface. Hence, the feature points on each patch are recovered independently of other patches, which is followed by interpolation of additional points using FFD, and therefore a dense point distribution can be obtained. At the end, a global smooth surface can be fit to the large number of points available from every patch.

4.4.2 Comparative Study

To validate my approach I compared its performance against two other approaches. One of them makes use of the upper-bound model and the other uses the differential model:

1. The SOCP-based algorithm proposed in [32], in which the authors use a triangular mesh together with a-priori datasets for learning the deformations. The deformation constraints are expressed using the upper-bound model.

2. The approach of [56], where the authors used a differential model to represent the deformation constraints, and applied an optimization method to this model.

In a previous approach [63], I have proposed the use of a ToF-camera in combination with a conventional camera in order to enhance the reconstruction when there is a limited number of tracked points. However, my new approach enhances the results of the previous approach when a large number of points are available. Table 4.1 shows that my approach gives good results and outperforms both other approaches, although the improvement is not significant (the numerical errors were calculated with respect to the ground-truth data). The improvement comes from the fact that the other approaches require the deformation to be very close to inextensible, whereas in my approach near-isometric deformations can be handled.

Camera Setup and Noise

: In all experiments, the 3D data is projected onto the images using 4 different camera setups, designed by varying the distance from the object to the camera and also the focal length, so that increasing levels of perspective distortion (Setup 1: $z = 200$, $f = 1000$; Setup 2: $z = 150$, $f = 1000$; Setup 3: $z = 100$, $f = 800$; Setup 4: $z = 50$, $f = 500$) can be obtained. I also show results for increasing levels of Gaussian noise added to the

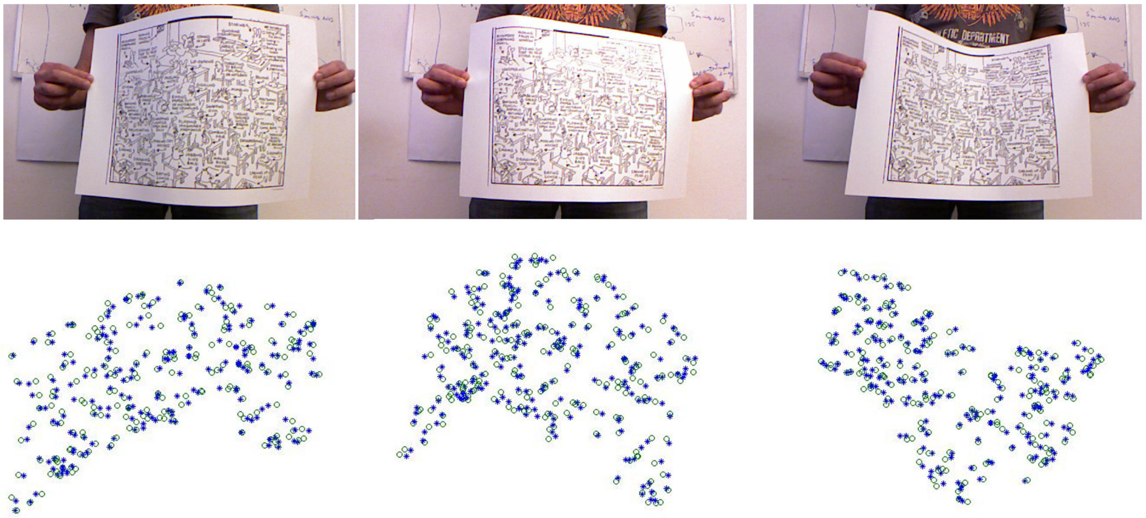

Figure 4.5: Sample real images with points in a random pattern. The points marked with '*' and those marked with 'o' show the estimated and the ground-truth points on the paper-sheet, respectively. The ground-truth was acquired using Kinect. The absolute reconstruction errors ($RE_{abs} = \frac{1}{N}\sum_{i=1}^{N} \|\mathbf{p}_i - \hat{\mathbf{p}}_i\|^2$) from the left to the right are 0.51, 0.42, 0.38 in cm.

coordinates of the image points, with the noise standard deviation varied from 0 to 3 pixels with 0.25 pixels increments. The reconstruction error can be seen in Figures 4.3 and 4.4, where the results correspond to averages obtained from 100 deformations, randomly selected. After having performed 10 trials for each deformation, each average value was acquired from 1000 trials. The estimated 3D positions, and their equivalent ground-truth from some frames of the cardboard, marked with '+' and 'o' respectively, are shown in Figure 4.2.

4.4.3 Experiments with Real Data

The algorithms used in the experiments with synthetic data were also applied to real data. This data consists of 2D point tracks from a sequence of images, recorded using a calibrated camera. Various objects, including the paper-sheet shown in Figure 4.1, were selected for tests with real data. A large number of inextensible deformations were created. Some of them and their reconstructed 3D points are illustrated in Figure 4.5, with the points being

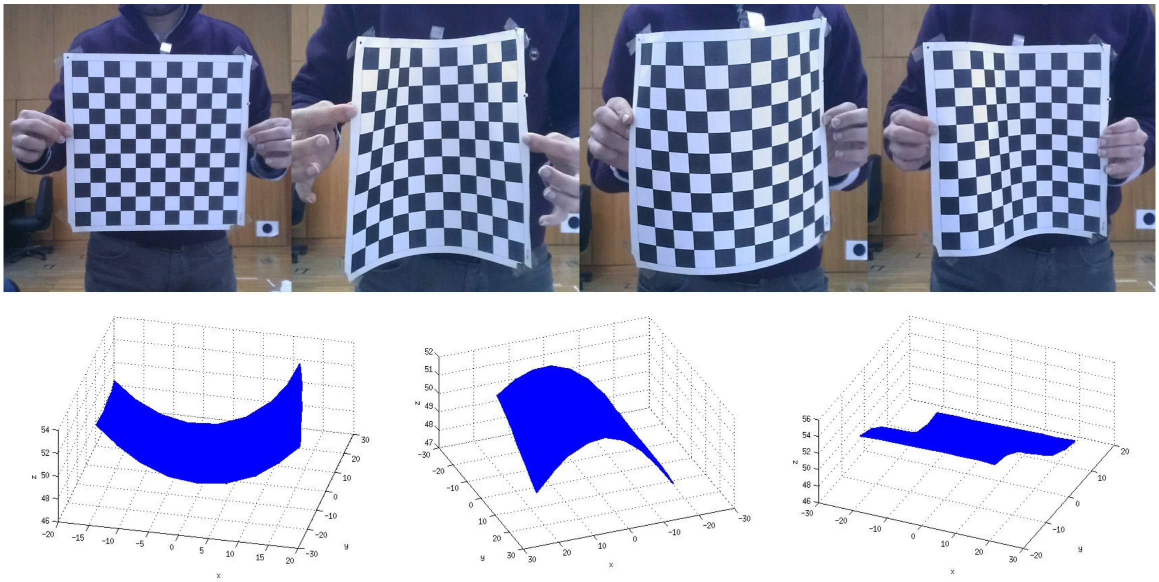

Figure 4.6: Top: The first image from the left is the template. Horizontal axis is the x-axis, vertical axis is the y-axis. Bottom: The reconstructed shapes of the above deformations.

in no fixed pattern. To show the quality and effectiveness of the approach under normal measurement conditions, we carried out a quantitative evaluation on the paper-sheet, given that its ground-truth is available. A total of five 6×6 patches were used. Obviously, the 2D tracks come with an unknown level of noise. In this case, the reconstruction error, obtained from 25 deformations, has been calculated to be 1.58e-04. More examples of deformations can be seen in Figure 4.6. The efficiency of the approach is visible from the recovered shapes.

4.5 Conclusions

In this subtask, monocular reconstruction of deformable, inextensible surfaces using one image was addressed. To perform this reconstruction, an SDP optimization was formulated by imposing constraints corresponding to the type of deformation i.e. deformation constraints. These constraints were defined using a differential model of the deformation, and also the upper-bound model. The application of the differential model requires that a parametric mapping between the template and the 3D surface be defined. I used a Free-

Form Deformation for this purpose. Both types of constraints were then combined in a SDP optimization program. The optimization estimated the 3D positions of the points on the deformed surface. The experimental results show and characterize the performance of the proposed approach.

Conformal Surface Reconstruction

4.6 Problem Statement

In addition to inextensible surfaces, here I address monocular reconstruction of surfaces that undergo extensions while being distorted by tools with a sharp point, as shown in Figure 4.9. Examples of such surfaces include plastic balloons, sports balls, abdominal tissues in laparoscopic surgery, etc. These extensions can be generally described by the deformation type called *conformal*, which is distinguishable from isometric deformation in that it preserves the angles between the curves on the surface, but not the distances. The existing solutions involve complex algorithms that usually have practical limitations rendering the implementation of these algorithms difficult. An analytical approach was developed for isometric and conformal deformations using Partial Differential Equations [47]. The approach was developed for weak-perspective projection and requires complex differential models. A SLAM method for elastic surfaces was developed, using fixed boundary conditions [50]. In [51], the authors formulate the reconstruction problem of a generic surface in terms of the minimization of stretching energy and impose a set of fixed boundary 3D points to constrain the solution. This approach deals with a general group of elastic surfaces without applying any constraints explicitly associated with conformal deformation. I aim at the development of simple and accurate models while emphasizing their feasibility in practice. Prior data consists of a 3D template, its image and n_p point correspondences (contained in vector $\mathbf{p}$) from a given image of the surface. The 3D template, in this case, is equivalent to the largest expanded shape considered for reconstruction. The goal is to compute the 3D positions of points **p** and their stretching factors. The deformed surface is represented by a parametric function based on Free-Form Deformations (FFD). This function maps the template onto the surface and it is used to analytically express what we call *conformal constraints*. These constraints correspond to the nature of conformal deformation. My approach is built on the assumption that the surface boundary undergoes no motion or short-range motion. In other words, as in [51], the positions of boundary points do not change significantly from image to image. Such points are determined depending on the camera viewpoint, as illustrated in Figure 4.9. To perform the reconstruction, we take into account four groups of constraints, namely, the reprojection error, upper-bound model, conformal constraints, and also those associated with the boundary points. All these constraints are integrated into a unified optimization procedure which is executed multiple times, with the upper-bound model being updated after each optimization. The updating implies that the bound on the Euclidean distance of pairs of points decreases. As a result, the optimization is repeated with the different bounds considered. The optimal reconstruction is finally determined by selecting the one that minimizes a specific set of criteria. The final results include estimates of the 3D positions of the points and a value for the stretching factor at each point.

Organization of this subtask: First, I describe the proposed approach, including all types of constraints employed in the optimization procedure. The optimization itself is described in detail in the next section. The experimental results are presented next. The last section contains conclusions.

4.7 Proposed Approach: Constraints

Modified Upper-Bound Model: In general, upper-bound model is adopted for the reconstruction of surfaces that deform isometrically and it is used in an optimization, along with the maximum depth heuristic and the minimization of the reprojection error, to estimate the 3D positions of points on the isometric surfaces. This optimization is implemented by means of a second-order-cone programme (SOCP) [33, 40].

Compared with the reconstruction of isometric surfaces, the reconstruction of conformal surfaces involves additional ambiguity/uncertainty, arising from extra unknowns, namely, the stretching factors. These factors have to be estimated apart from the positions of points $\mathbf{p}$. Therefore further information and/or constraints must be considered so that all unknowns, including the stretching factors, can be determined. To cope with the added ambiguity, a variational form of the upper-bound model that allows for a wider variety of deformations, extensible for example, will be constructed. In this case, the modified extensibility constraint $\left\|\mathbf{p}_j - \mathbf{p}_k\right\| \leq \lambda.d_{jk}$ is used. This constraint varies depending on the parameter λ which, as will be shown later, decreases (at small regular intervals specified in advance). This variational formulation leads to the estimation of a set of possible surface deformations.

Conformal Constraints:

Theorem: A surface $\mathbf{S}$ is conformal or angle-preserving if and only if the coefficients of First Fundamental Form (FFF) are proportional, i.e., $\mathbf{I} = \eta \begin{pmatrix} u, & v \end{pmatrix} \mathbf{I}^*$ for some scalar function $\eta \neq 0$, referred to as stretching factor. As shown earlier, for isometric surfaces, the above proportional equation is simplified to $\mathbf{I} = \mathbf{I}^*$.

The 3D template enables the calculation of the FFF of the surface at every point. Hence, $\mathbf{I}$ is known in advance. As stated earlier, the coefficients of $\mathbf{I}^{'}$ can be expressed in terms of $\mathbf{p}$ - refer to Equation 4.4. The goal is to estimate vector $\mathbf{p}$, containing all the points. As a result, for the point i, we have:

$$E_i^{'}(\mathbf{p}_i) = \eta_i.E_i, \;\; G_i^{'}(\mathbf{p}_i) = \eta_i.G_i, \;\; F_i^{'}(\mathbf{p}_i) = \eta_i.F_i \tag{4.16}$$

Again, as in Equation 4.5, we derive another quadratic expression so that $F^{'}$ can be replaced by the subsidiary function $H^{'}$, which is quadratic, similarly to $E^{'}$ and $G^{'}$:

$$H_i^{'}(\mathbf{p}_i) = \eta_i.(E_i + G_i + 2.F_i) = \eta_i.H_i \tag{4.17}$$

Constraints from Boundary Points: These constraints are defined as quadratic cones:

$$\left\| p_{s,x} - p_{s,x}^{\circ} \right\| \leq \varepsilon_x,\ \left\| p_{s,y} - p_{s,y}^{\circ} \right\| \leq \varepsilon_y,\ \left\| p_{s,z} - p_{s,z}^{\circ} \right\| \leq \varepsilon_z \tag{4.18}$$

where $s \in \left(1, \ldots, n_f\right)$ and n_f indicates the number of boundary points, restricted to a motion within a range defined by ε_x, ε_y, ε_z as well as $p_{s,x}^{\circ}$, $p_{s,y}^{\circ}$ $p_{s,z}^{\circ}$, which specify the x, y and z coordinates of the boundary point s on the template. The boundary points are included in the vector $\mathbf{p}$ as part of the points considered for reconstruction. Also, we exclude any point pairs linking only boundary points.

Reprojection Error: Consider that $\mathbf{K}$ is a known calibration matrix with focal length f and that $\mathbf{q}^{'} = (\ u^{'},\ v^{'}\ )$ denotes the image points - corresponding to $\mathbf{p}$ - detected on the image of the deformed surface. For point i, the projection equations are defined as $f.p_{x,i} - u_i^{'}.p_{z,i} = 0$ and $f.p_{y,i} - v_i^{'}.p_{z,i} = 0$.

4.8 Proposed Approach: Optimization

The four types of constraints discussed above are joined in definition of an optimization procedure aimed at computing both the 3D positions of points $\mathbf{p}$ and the stretching factors η_i. The 3D template, as previously described, denotes the most expanded size of the surface to be reconstructed. In this case, the coefficients of FFF have the biggest values, while the values of these coefficients for the other sizes of the surface will be, in general, smaller.

Therefore, it follows that the values for η_i will be within the range 0 to 1.

The equalities in Equations 4.16 and 4.17 can not be treated as a regular constraint in view of the fact that a. Either side of them should be constant, which is not valid due to the variables η_i; b. They are non-convex and inconsistent with the other constraints. Instead, we define quadratic cones, as shown in Equation 4.19, and later propose an effective technique for dealing with these equalities. The new set of quadratic cones is defined as follows:

$$\left(E_i^{'}(\mathbf{p}_i)\right)^{\frac{1}{2}} \leq E_i^{\frac{1}{2}},\ \left(G_i^{'}(\mathbf{p}_i)\right)^{\frac{1}{2}} \leq G_i^{\frac{1}{2}},\ \left(H_i^{'}(\mathbf{p}_i)\right)^{\frac{1}{2}} \leq H_i^{\frac{1}{2}} \tag{4.19}$$

Let us express the re-projection error as: $e_{k,u} = f.p_{x,k} - u_k^{'}.p_{z,k}$ and $e_{k,v} = f.p_{y,k} - u_k^{'}.p_{z,k}$ and assume that $p_{z,k}$ denotes the depth of the points. Then, an optimization procedure that allows the reconstruction of the surface can be formulated, using SOCP, as:

$$\min_{\mathbf{p}} \sum_{i=1}^{np} \left(-p_{z,i} + w.\left(\left(e_{i,u}\right)^2 + \left(e_{i,v}\right)^2\right)\right),\ \text{subject to}$$

Constraints in Equations 4.18 and 4.19,

$$\left\|\mathbf{p}_j - \mathbf{p}_k\right\| \leq \lambda.d_{jk}\ \text{for every selected pair}\ j,k \tag{4.20}$$

where $s \in \left(1,\ldots,n_f\right)$ and w is a scalar used to specify a weighted summation and is tuned empirically. The above optimization is repeated, with the value for λ decreasing each time (the maximum value is 1.). After each iteration, both the reprojection error and the values for

$$\eta_{i,E} = E_i^{'}(\mathbf{p}_i)/E_i, \;\; \eta_{i,G} = G_i^{'}(\mathbf{p}_i)/G_i, \;\; \eta_{i,H} = H_i^{'}(\mathbf{p}_i)/H_i$$

$$e\eta = \frac{1}{np}\sum_{i=1}^{np}\left[(\eta_{i,E}-\eta_{i,G})^2+(\eta_{i,E}-\eta_{i,H})^2\right] \tag{4.21}$$

are recorded. The iterations stop when λ reaches a pre-defined value (which is near zero.). The results corresponding to the optimization where the smallest value for the sum of the two error criteria (reprojection error and $e\eta$) has been obtained are selected as the best reconstruction. The stretching factors are then computed as:

$$\eta_i = \left(\eta_{i,E}+\eta_{i,G}+\eta_{i,H}\right)/3 \tag{4.22}$$

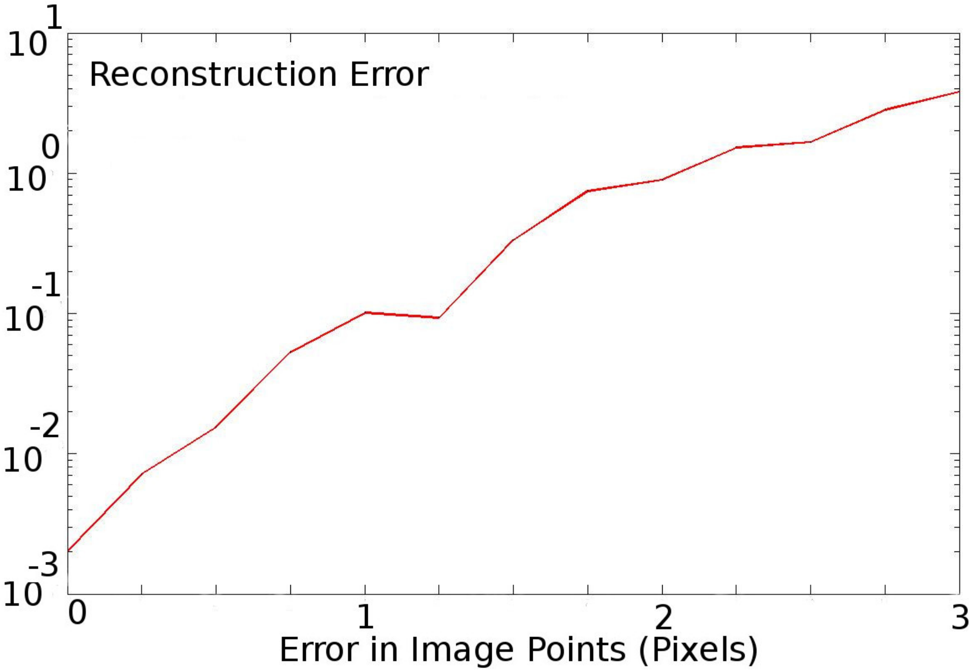

Figure 4.7: Reconstruction error for increasing levels of noise.

4.9 Experimental Results

Synthetic Data: To validate the efficiency of the proposed approach it was evaluated by conducting several experiments. This evaluation requires the availability of data from surfaces that deform conformally. The ground-truth synthetic data to be used in the experiments must also correspond to a conformal deformation. To generate this data I used a 3D graphics software. Two samples of such deformations are shown in Figure 4.8, which also meet the requirements for having boundary points not moving or deforming. These points can be specified by marking the margin surrounding the deformed part. In Figure 4.8, they correspond to the upper part of the first top layer of the 2 shapes, given that the camera is viewing the surface from below and with the optical axis aligned with the Z axis. The boundary points of an object depend on the camera viewpoint. Point correspondences were determined between the images of the deformed shapes and the template image. We consider an extended shape as the template. To calculate the geodesics between any selected pair of points, the template is flattened out by the use of conformal flattening. Each input image is computed by projecting the 3D points on the deformed surface onto an image plane, taking into account a virtual camera defined by a specific calibration matrix.

I show results for increasing levels of Gaussian noise added to the coordinates of image points. The standard deviation of the noise was varied from 0 to 3 pixels with 0.25-pixel increments. The accuracy of the results is characterized by means of a reconstruction error which is defined as the normalized Euclidean distance between the observed ($\hat{\mathbf{p}}_i$) and the estimated ($\mathbf{p}_i$) world points defined as $RE = \frac{1}{N} \sum_{i=1}^{N} \left[\|\mathbf{p}_i - \hat{\mathbf{p}}_i\|^2 \,/\, \|\hat{\mathbf{p}}_i\|^2 \right]$. The reconstruction error is shown in Figure 4.7, whose values correspond to averages obtained from 10 deformations randomly generated. Since 50 repetitions were performed for each deformation, each average value was obtained from the values of from 250 trials. The quantitative results demonstrate that the approach performs efficiently even in the presence of noise. See Figure 4.8 for some qualitative results.

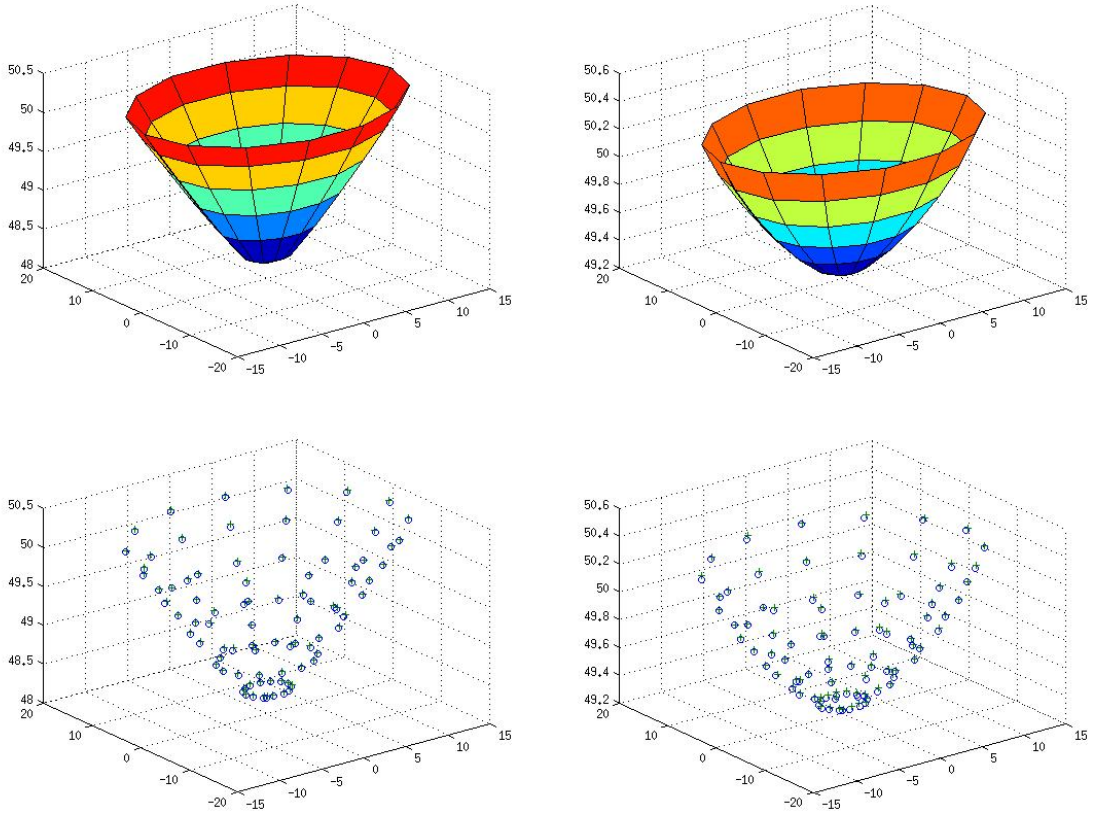

Figure 4.8: Top: a surface that extends while its boundary points have small or no motion. These points are on the margin of the upper part of the surface, and are defined by considering the camera viewpoint. The left deformation corresponds to a higher degree of stretching than the right one. The virtual camera is located under the surface. Bottom: reconstruction of the deformations displayed on top; The points marked with '+' and those marked with 'o' illustrate the computed and the ground-truth points, respectively.

I also made performance comparisons between my approach and the one proposed in [51]. I attempted to implement the general structure of this approach. My approach is similar to this approach in that I use also constraints on the motion of the boundary points. However, in my case, we explicitly consider constraints resulting from conformal deformations. My approach performs better in case of larger extensions. The other approach addresses the reconstruction problem using Mechanics-based modeling and uses a constraint based on the stretching energy of the surface. It does not consider conformal-related constraints and yields good results in case of smaller extensions. In the experiments that I performed I obtained an overall reconstruction error 0f 0.002 with my approach, whereas the other approach had a reconstruction error of 0.006 in 3D distance measurements.

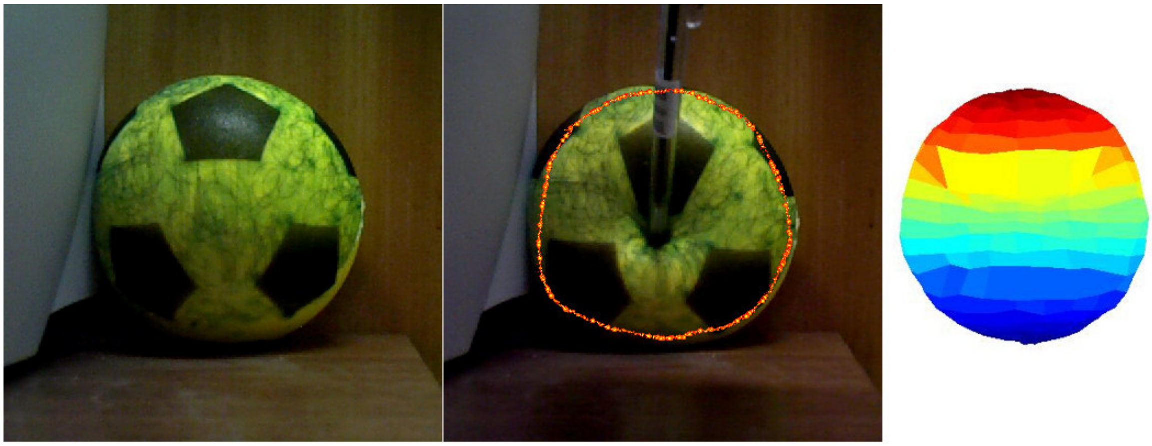

Figure 4.9: A ball with approximately conformal deformation. The red circular line indicates the location of the boundary points. The right-hand image is the reconstructed model of the deformed ball.

Real Data: Since the approach described in this subtask deals with the reconstruction of surfaces undergoing conformal deformation, to obtain real data we needed an object that deforms (at least approximately) according to the conditions of conformal deformation. For that purpose, we chose balls and plastic balloons as objects necessary to acquire the data. The algorithms used with synthetic data were applied to images acquired by a calibrated camera. An example image is given in Figure 4.9. As shown in this figure, the boundary points nearly do not move. However, the deformation does not seem to comply with the conditions of a strictly conformal deformation. Features were obtained manually or semi-automatically with SIFT, that is, using interactive point selection with initial matches from SIFT. The sample result shows that the presented method results in fairly accurate reconstructions.

4.10 Conclusions

The preceding work was an investigation of conformal deformation reconstruction, aiming to propose an easy-to-implement methodology that still attains accurate reconstructions.

The results show that the approach is reasonably efficient. Assuming that the boundaries/contours of the surface undergo negligible motion, a set of estimates was obtained for the 3D positions of points on the surface using a modified, variational formulation of the upper-bound model and The best estimate was then selected based on some criteria.

CHAPTER 5

GEOMETRIC SURFACE TRANSFORMATIONS AS DEFORMATION MODELS

5.1 Introduction

In this chapter, I tackle the third task and investigate the use of surface transformations for reconstruction purposes. This task comprises two subtasks.

The first subtask is to deal with the reconstruction of surfaces that deform under a variety of conditions. The deformation can range from no extension to a certain degree of extensibility. To perform the reconstruction, I use a single image and a 3D reference image of the surface, which can correspond to its undeformed state. The 3D reference image can be computed by any appropriate method. In particular, and in my case, I use homographies defined from two views of the surface. To proceed with the 3D reconstruction of the deformed surface I assume that the deformations are locally homogeneous and that the overall surface deformation can be obtained by combining the local deformations. For that purpose the surface is split into small patches. For each patch a mapping between the undeformed and the deformed shape is computed. The mapping is specified by using the quadratic deformation model [64]. As a result, given the undeformed shape, I define an optimization procedure whose goal is to estimate the 3D positions of deformed points in each image. The optimization operates on each patch, independently of the others. The experimental results show that this approach allows the precise reconstruction of a wide class of real deformations.

In the second subtask, I only deal with the monocular reconstruction of extensible surfaces by proposing a novel approach for the determination of the 3D positions of a set of points on images of a deformed surface. Given a 3D template, this approach is applied to each image independently of the others. To proceed with the reconstruction, the surface is divided into small patches that overlap in chain-like form. I model these surface patches as being uniformly extensible. Using a linear mapping from the template onto a patch, the variation of the patch shape is split into a rigid body transformation and a pure deformation. To estimate the pure deformation, I use an optimization procedure that minimizes the reprojection error along with the error over a constraint associated with uniform expansion. Having estimated the pure deformation, the rigid body transformation can be determined by decomposing the essential matrix between the current image and the virtual image that results from projecting the 3D positions that correspond to pure deformation of the template. This enables complete estimation of the linear mapping, thereby obtaining the 3D positions of the surface patch up to scale. To define a common scale, the surface smoothness is enforced by considering that the overlapping points of the patches are the same. The experimental results show the feasibility of the approach and that the accuracy of the reconstruction is good.

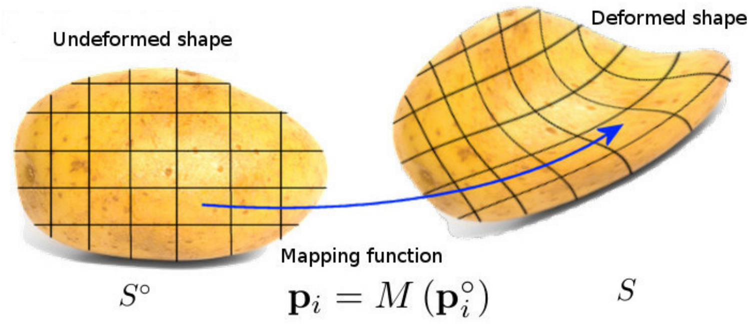

Figure 5.1: Transformation between the undeformed and the deformed shape.

Reconstructions Based on Homogeneous Deformation

5.2 Problem Statement and Model

In this subtask, I deal with the problem of reconstructing deformable surfaces from a single view. The type of deformations I take into account is not limited to inextensible surfaces but covers a larger group of surfaces that can undergo extension and/or compression. I assume that the surface to be reconstructed tends to change shape gently over time and is represented by a specific number of points/features lying on it. Given a set of feature/point correspondences from images acquired by a camera, we aim at estimating the 3D positions of the points through a general approach that is capable of handling a wide range of deformations, whether they are inextensible or not. In fact, this approach should permit the reconstruction results to be comparable to those of approaches intended for a specific deformation type, such as inextensible, conformal or elastic. To achieve this, the problem needs to be constrained adequately according to the deformation behaviour of the surface, in addition to imposing other conditions, namely the projection model. To this end, inspired by kinematics of deformable solids, we make use of the observation that the manner in which any natural surface deforms smoothly can be described by a combination of locally homogeneous deformations. However, it is not clear which subregions of the surface conform to such a deformation. Therefore, the subregions with purely homogeneous deformation and their borders are not known. In addition, a determination of these sub-regions requires segmentation, which is beyond the scope of this work. Consequently, we divide the surface into approximately equal patches, that are small enough to allow the assumption that they undergo near-homogeneous behaviour. To reconstruct each patch, we exploit the fact that a homogeneous deformation can be represented by means of a linear transformation between an undeformed and the deformed shape. Since the patches are approximately homogeneous, a more general mapping that is able to characterize a variety of deformation types is applied. This mapping consists in the quadratic deformation model, proposed in [64]. Figure 5.1 illustrates how the surface undergoes a change of shape, which can be described by this model. For each patch, the parameters related to the linear component of this quadratic transformation are dominant with respect to the other parameters. For this reason, I propose a constraint whose effect in the estimation is to approximate the quadratic transformation by a linear transformation that forces the linearization of the quadratic transformation. To compute the undeformed shape several methods could be used. In this case, I use an approach based on local homographies. Finally, by combining the data reprojection error with the constraints associated with homogeneous deformation, we arrive at an optimization scheme to determine the 3D positions of the points, thereby recovering the deformed shape.

Organization of this subtask: In the first section, I present an overview of the deformation gradient, the quadratic deformation model as well as the homogeneous deformation. Then, I discuss in detail the proposed approach, whose constituent parts include: a homography-based rigid reconstruction for computing the undeformed shape, the constraints derived from homogeneous deformation behavior, and an optimization scheme to determine the deformed shape. Next, I describe the experiments performed and demonstrate the results. The last section presents conclusions.

5.3 Background

Deformation Gradient: A deformation can be quantified and specified by the point-to-point mapping function $\mathbf{p}_i = M\left(\mathbf{p}_i^\circ\right)$ from the undeformed shape S° onto the deformed shape S, as shown in Figure 5.1. The deformation gradient is the fundamental measure of deformation in continuum mechanics. It is the second order tensor which maps line elements on S° into line elements (consisting of the same material particles) on S. Let a point i on S° be defined as $\mathbf{p}_i^\circ = \begin{bmatrix} x_i^\circ & y_i^\circ & z_i^\circ \end{bmatrix}^T$ and the corresponding point on S as $\mathbf{p}_i = \begin{bmatrix} x_i & y_i & z_i \end{bmatrix}^T$, the deformation gradient $\mathbf{F}$ is then:

$$\mathbf{F} = \begin{bmatrix} \frac{\delta x_i}{\delta x_i^\circ} & \frac{\delta x_i}{\delta y_i^\circ} & \frac{\delta x_i}{\delta z_i^\circ} \\ \frac{\delta y_i}{\delta x_i^\circ} & \frac{\delta y_i}{\delta y_i^\circ} & \frac{\delta y_i}{\delta z_i^\circ} \\ \frac{\delta z_i}{\delta x_i^\circ} & \frac{\delta z_i}{\delta y_i^\circ} & \frac{\delta z_i}{\delta z_i^\circ} \end{bmatrix} \tag{5.1}$$

Quadratic Deformation Model: According to [64], the quadratic deformation model for non-rigid bodies, which is composed of the rigid shape matrix and quadratic and cross-term components, can account for any deformations of the object. In fact, it provides a general representation of deformation. For point i, this model is defined by $\mathbf{p}_i = M_1\left(\mathbf{p}^{\circ}\right) = \mathbf{Q}.\mathbf{s}_i + \mathbf{c}$ where $\mathbf{c}$ is a 3×1 translation vector and

$$\mathbf{Q} = \left[\begin{array}{ccc} \mathbf{Q}_1 & \mathbf{Q}_2 & \mathbf{Q}_3 \end{array}\right]_{3\times 9}, \quad \mathbf{s}_i = \left[\begin{array}{ccccccccc} x_i^{\circ} & y_i^{\circ} & z_i^{\circ} & x_i^{\circ 2} & y_i^{\circ 2} & z_i^{\circ 2} & x_i^{\circ} y_i^{\circ} & y_i^{\circ} z_i^{\circ} & z_i^{\circ} x_i^{\circ} \end{array}\right]^T \tag{5.2}$$

$\mathbf{Q}_1$, $\mathbf{Q}_2$ and $\mathbf{Q}_3$ are 3×3 transformation matrices associated respectively with the linear, quadratic and cross-term deformations at a given frame. Applying the quadratic transformation to the shape matrix $_i$ we obtain the 3D position of point $\mathbf{p}_i$ at the frame.

Homogeneous Deformation: A *homogeneous deformation* is one where the deformation gradient is uniform, i.e. independent of the coordinates. The respective transformation is then defined as:

$$\hat{\mathbf{p}}_i = M_2\left(\mathbf{p}^{\circ}\right) = \mathbf{Q}_1.\mathbf{p}_i^{\circ} + \mathbf{c} \tag{5.3}$$

where $\mathbf{c}$ is a 3×1 vector that denotes the rigid body translation. This transformation is viewed as a special case of the quadratic deformation model.

5.4 Proposed Approach

As stated earlier, this work is aimed at developing a generalized model to describe and estimate the deformation of a variety of deforming surfaces without considering a specialized algorithm for each particular type of deformation. For this purpose, in view of the fact that any smooth deformation can be made up of some smaller homogeneous deformations, we partition a typical surface into patches whose deformation is approximately homogeneous. The goal is to ensure that they show such deformation, independently of one another. This could be represented mathematically by what we call *homogeneous deformation constraints*. These constraints serve to force $M_1(.)$ to be close to $M_2(.)$ such that the linear parameters make up the major component of $M_1(.)$. Deriving such constraints requires the 3D positions of the points $\mathbf{p}_i^\circ$ to be known. A rigid reconstruction method is thus described in order to recover the undeformed shape. By combining the deformation constraints with the data reprojection error, an optimization procedure is then formulated for determining the 3D positions of the deformed points $\mathbf{p}_i$.

Computation of the Undeformed Shape: The underformed shape or reference shape can be computed by a variety of methods. In particular I adopt a practical method based on homographies which requires two images of the surface without deformation, as represented in Algorithm 1. The following is a detailed description of this rigid reconstruction.

Assuming that the surface is not in motion and remains in the undeformed state for a while, the camera starts capturing the scene and surface while moving. A selection of frames that cover different fields of view is then made. One of them is adopted as base image. We compose image pairs, each consisting of the base image plus another from the frame selection, called support image. In addition, the points $\mathbf{p}_i^\circ$ are randomly grouped into distinct sets, each of which includes 3 points and can share no points with other sets. An estimate of the undeformed shape could be found per image pair through the following 2-view rigid reconstruction algorithm:

Step 1- The essential matrix between the base image and the support image is determined by using the image points in one image and the corresponding points in the other.

Step 2- By decomposing the essential matrix, we can compute the rotation $\mathbf{R}$ and a scaled translation $\bar{\mathbf{t}}$ between the 2 views. As indicated in [65], the decomposition yields four combinations of $\mathbf{R}$ and $\bar{\mathbf{t}}$, one of which is feasible. Positive depth constraint is then used to disambiguate the physically impossible solutions. Let us assume that $\mathbf{R}$ and $\bar{\mathbf{t}}$

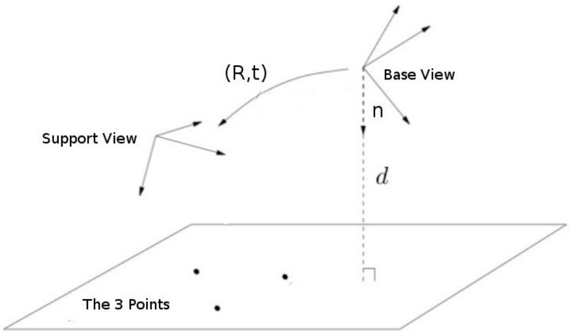

Figure 5.2: Homography between the base and support views

contain the correct values.

Step 3- Each 3-point set constitutes a plane, for which an scaled Homography can be determined according to Figure 5.2:

$$\mathbf{H} = \mathbf{R} - \frac{\bar{\mathbf{t}}.\mathbf{n}^T}{d} \tag{5.4}$$

where $\mathbf{n}$ and d denotes the normal to the plane and the distance from the origin, respectively. As a consequence, in order to define the equation of the plane and eventually estimate the 3D positions of the 3 points, $\mathbf{n}$ and d should be known. d can be, however, eliminated by setting it to 1. This way leads to an extra re-scaling of the positions without loss of generality. We now build a linear system of 3 equations (one for each point), by which we may achieve a correct analytical solution using the normal equation, thus estimating the normal vector $\mathbf{n}$, which consists in $\mathbf{n}^T = \begin{bmatrix} n_x & n_y & n_z \end{bmatrix}$. Consider that for one of the 3 points, $\mathbf{q}_b^\circ$ and $\mathbf{q}_s^\circ$ are 3-vectors corresponding to its homogeneous image points on the base and the support image, respectively. We will therefore have

$$\lambda.\mathbf{q}_s^\circ = \mathbf{H}.\mathbf{q}_b^\circ \tag{5.5}$$

where λ is the ratio of the point depth from support view to that from base image. The 3 rows of the Equation 5.5 are linearly dependent. However, we just need one equation per point that is only a linear function of n_x, n_y and n_z. Such an equation could be derived by simply replacing λ in either the first or second row by the last row. The same procedure can apply to the other 2 points, thereby resulting in a 3-equation linear system, where the only unknowns are n_x, n_y and n_z. This system has a single exact solution that can be obtained via the normal equation. Having estimated the normal vector, it is trivial to calculate the 3D positions of the points in the coordinate system of base view by joining the projection model with the plane equation.

Step 4- The step 4 is performed for all the 3-point sets of the image pair to determine all the 3D positions.

Step 5- The steps 1 to 5 are repeated with the other image pairs to make additional 3D estimates.

Step 6- Finally, we compute the average of all the estimates from all image pairs to attain a reliable 3D structure for the undeformed shape with respect to the local referential of base view. Note that, this is just a scaling of the true structure.

Homogeneous Deformation Constraints: Given that the undeformed shape has been computed, and considering an image of the deformed surface, the objective is to determine the 3D positions of the points.

The set of deformation constraints is defined so that, in the case of near-homogeneous deformations, the linear parameters $\mathbf{Q}_1$ determine the mapping $M_1(.)$, i.e. they are more significant than the quadratic and cross-term parameters $\mathbf{Q}_2$ and $\mathbf{Q}_3$. These constraints are composed of the following: 1- the gradient of the transformation $M_1(.)$ at each point $\mathbf{p}_i$ with respect to $\mathbf{p}_i^\circ$ should be approximately the same. 2- Conic constraints are used to constrain the positions of the points $\mathbf{p}$ to a specified region around the points $\hat{\mathbf{p}}$. As a result, the deformation constraints for point i, are defined as:

Algorithm 1: Rigid Reconstruction

```
Input: $\mathbf{q}_b^\circ$ and $\mathbf{q}_s^\circ$, Output: $\mathbf{p}^\circ$.
* The surface remains still. A rotating camera captures its image.
* Make a selection of 2 frames covering different fields of view. Call either of these two frames base image and the other support image
* Arrange points $\mathbf{p}_i^\circ$ randomly into distinct sets, each of which includes 3 points. Preferably these sets do not have any points in common.
while There are still 3-point sets to be processed do
    * Pick out one of the sets.
    * Compute the essential matrix between the base image and the support image
    * Decompose the essential matrix to estimate the rotation $\mathbf{R}$ and a scaled translation $\bar{\mathbf{t}}$ between the 2 views.
    * Define a Homography for the plane passing through the 3-point set as:
    $\mathbf{H} = \mathbf{R} - \frac{\bar{\mathbf{t}}.\mathbf{n}^T}{d}$ where $d$ is a scalar as the distance and $n$. the plane normal
    * Compose a 3-equation linear system from the following formula such that its unknown is the normal vector: $\lambda.\mathbf{q}_s^\circ = \mathbf{H}.\mathbf{q}_b^\circ$
    * Estimate the normal vector and then calculate the 3D positions of the 3 points in the coordinate system of base view by combining the projection model with the plane equation.
end while
```

$$\mathbf{F}_i(M_1(.)) \cong \text{constant}, \;\; 1 \le i \le n$$

$$\left\|\mathbf{p}_{i,x} - \hat{\mathbf{p}}_{i,x}\right\| \le \delta_x, \;\; \left\|\mathbf{p}_{i,y} - \hat{\mathbf{p}}_{i,y}\right\| \le \delta_y, \;\; \left\|\mathbf{p}_{i,z} - \hat{\mathbf{p}}_{i,x}\right\| \le \delta_z, \;\; 1 \le i \le n \tag{5.6}$$

where n is the number of points on the patch and $\mathbf{F}_i$ denotes the gradient associated with point i. Subscripts x, y, and z denote the x, y, z-coordinates and δ_x, δ_y, δ_z are thresholds to constrain the positions of the estimated points.

Estimation of the Deformed Shape: The constraints described above are used in an optimization procedure designed to minimize the regular data fitting residuals, namely the reprojection error. The goal of this minimization is to obtain an accurate estimate of the transformation matrix $\mathbf{Q}$ and of $\mathbf{c}$, therefore allowing the estimation of the 3D positions of the points $\mathbf{p}_i$ lying on the patch, by means of the application of the quadratic deformation model $M_1\left(\mathbf{p}_i^{\circ}\right)=\mathbf{Q}.\mathbf{s}_i$. The image projection model considered is weak-perspective. The use of the weak-perspective model leads to a simpler and more efficient formulation of reprojection error. Consider that $\mathbf{q}_i=\left[\begin{array}{cc} u_i & v_i \end{array}\right]^T$ are the image points corresponding to $\mathbf{p}_i$, the optimization problem can be expressed as a second-order cone program:

$$\min_{\mathbf{Q},\,\mathbf{c}} \sum_{i=1}^{n}\left[\left(Z_{ave}.u_i-f.M_1\left(\mathbf{p}_i^{\circ}\right)^{[1]}\right)^2+\left(Z_{ave}.v_i-f.M_1\left(\mathbf{p}_i^{\circ}\right)^{[2]}\right)^2+\right.$$

$$\left. w.\left(\mathbf{F}_1(M_1(.))-\mathbf{F}_i(M_1(.))\right)^2\right],\ \text{subject to}$$

$$\left\|\mathbf{p}_{i,x}-\hat{\mathbf{p}}_{i,x}\right\|\leq\delta_x,\ \left\|\mathbf{p}_{i,y}-\hat{\mathbf{p}}_{i,y}\right\|\leq\delta_y,\ \left\|\mathbf{p}_{i,z}-\hat{\mathbf{p}}_{i,x}\right\|\leq\delta_z,\ 1\leq i\leq n$$

$$\left\|\mathbf{p}_{i,x}-\mathbf{p}_{i,x}^{\circ}\right\|\leq\eta_x,\ \left\|\mathbf{p}_{i,y}-\mathbf{p}_{i,y}^{\circ}\right\|\leq\eta_y,\ \left\|\mathbf{p}_{i,z}-\mathbf{p}_{i,x}^{\circ}\right\|\leq\eta_z,\ 1\leq i\leq n \qquad (5.7)$$

where superscripts [1] and [2] indicate the first and second entry of the 3-vector $M_1\left(\mathbf{p}_i^{\circ}\right)$, respectively. f is the focal length and Z_{ave} the average depth of the points on the patch, specified by the mapping M_1. w is a factor to create a weighted sum of reprojection error plus a regularization term.

For deformations in which the deviation from the undeformed state is small, the latter set of conic constraints restricts the extent of position variation of the points with respect to the undeformed shape. Those constraints are specified by the thresholds η_x, η_y, η_z which are empirically estimated and which are, in general, bigger than δ_x, δ_y, δ_z. Refer to Algorithm 2 for an algorithmic representation of the non-rigid reconstruction described above.

Algorithm 2: Non-rigid Reconstruction

```
Input: $\mathbf{p}^{\circ}$, Output: $\mathbf{p}$.
* The undeformed shape was estimated. Consider an image of the deformed surface.
* Divide the surface into patches.
while there are still patches to reconstruct in the current frame do
    * Draw a random patch from the remaining ones.
    * Solve the optimization problem expressed in Equation 5.7 to minimize the error
    over deformation constraints along with the reprojection error.
    * Substitute the output of the optimization $\mathbf{Q}$ and $\mathbf{c}$ into the $\mathbf{p}_i = M_1\left(\mathbf{p}^{\circ}\right) = \mathbf{Q}.\mathbf{s}_i + \mathbf{c}$.
    At this point, the deformed positions are calculated.
end while
```

5.5 Experiments and Results

We have conducted several experiments with both synthetic and real data in order to implement and analyse the proposed reconstruction procedure.

Synthetic Data: The synthetic data consists in several 3D points distributed across the surface of an analytical 3D shape. The data is generated so that different types of deformation can be applied to the surface. These deformations can be inextensible or undergo expansions/contractions.

In the case of inextensible deformations, I use a 3D cloth simulator- see Figure 5.3. I also consider shapes that can be modeled analytically namely cylinders, ellipsoids and spheres-see Figure 5.4. Their analytical models can be used to generate the set of n point correspondences required. For example, the sphere is expanded/contracted uniformly by changing its radius. Although these volumetric shapes are simple surfaces, they are however adequate for synthetic testing so that I can evaluate our approach quantitatively. Points $\mathbf{p}$ are assumed to be visible.

The synthetic data corresponds to images acquired by a camera located 50 cm away from the surface, with a focal length of 500 pixels. In case of the undeformed/reference shape, the 3D data is rotated and translated with known values. In case of the deformed

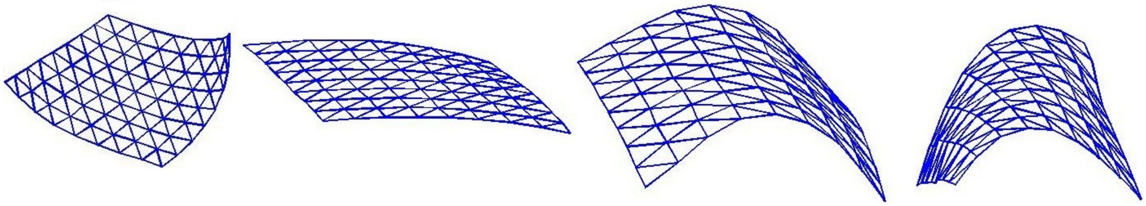

Figure 5.3: Inextensible Surface: Simulated deformations.

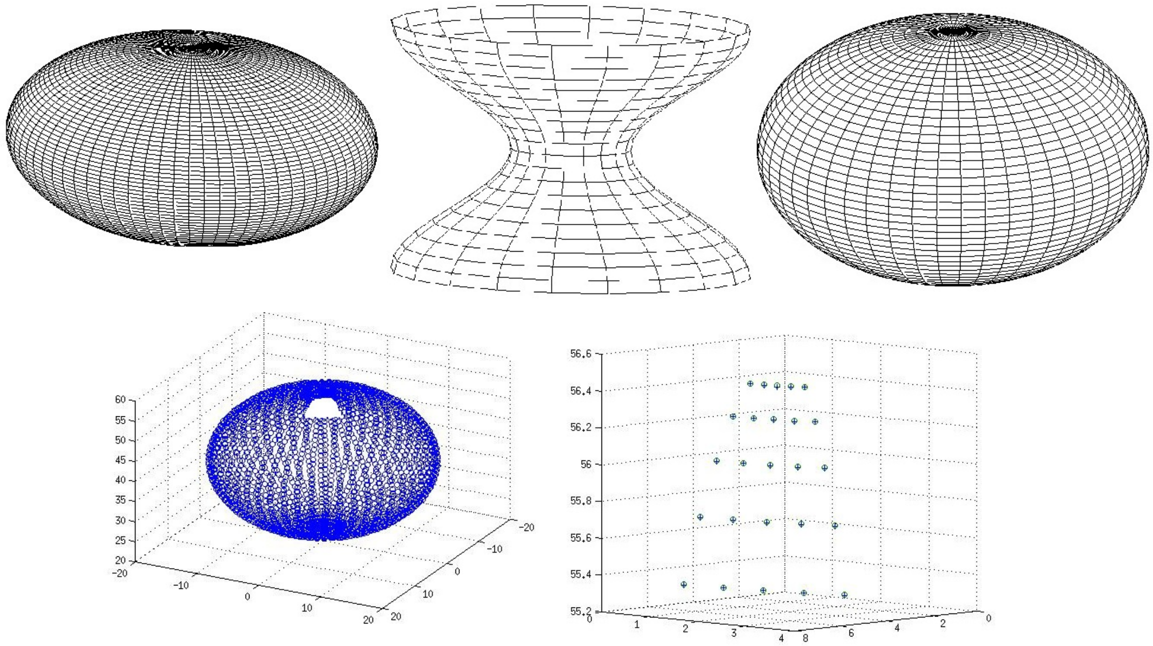

Figure 5.4: Synthetic extensible surfaces - Top: from left to right: ellipsoid, deformed cylinder, sphere. Bottom: synthetic extensible surface, the left image represents a patch of the sphere. The right image shows the points. The reconstructed and ground-truth points have been marked with '+' and 'o', respectively.

shape, the image is split into approximately identical patches. In my case I considered a maximum of 12 patches. Reconstruction was also performed separately for divisions with a smaller number of patches, The accuracy of the approach is evaluated by using the 3D reconstruction error, which is defined by the equation below.

$$RE = \frac{1}{N}\sum_{i=1}^{N}\left[\|\mathbf{p}_i - \hat{\mathbf{p}}_i\|^2 \,/\, \|\hat{\mathbf{p}}_i\|^2\right]. \tag{5.8}$$

This equation corresponds to the normalized Euclidean distance between the observed

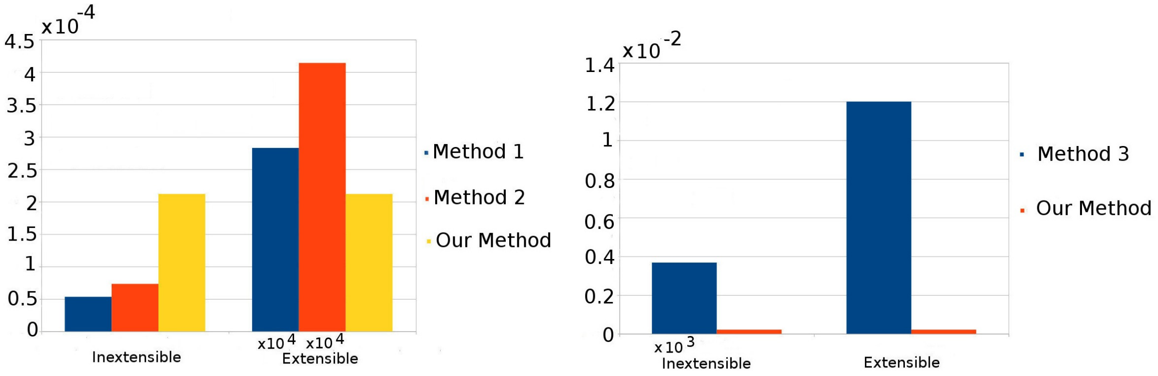

Figure 5.5: Comparison of reconstruction error between my approach and other methods. Two inextensible-case methods (Left) and one extensible-case method (Right), respectively. The results of my method correspond to the combined results for both deformation types. $\times 10^3$ and $\times 10^4$ indicate that the values are rescaled by $\times 10^3$ and $\times 10^3$, respectively.

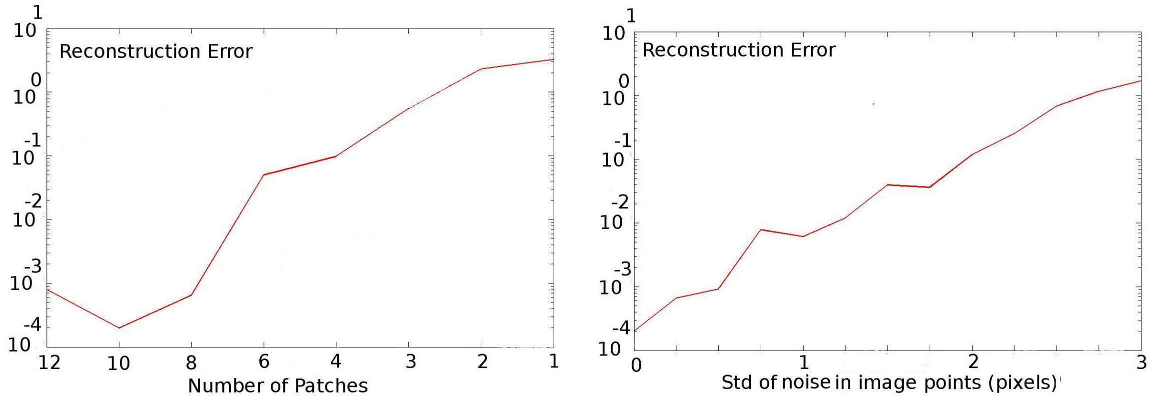

Figure 5.6: Left: Reconstruction error relative to the number of patches. Right: Reconstruction error relative to the noise in image points.

($\hat{\mathbf{p}}_i$) and the estimated ($\mathbf{p}_i$) coordinates of the world points, with N being the total number of points- the average N in my case is 90. Note that the results correspond to average values obtained by iterating the optimization 100 times for each of the 10 deformations considered. Taking into account that the reconstruction is performed up to a scale factor, an estimate for the scale factor has to be computed. The left-hand plot of Figure 5.6 shows the variation of the reconstruction error as a function of the number of patches. In addition to the case of data without noise I also performed the 3D reconstruction in cases where Gaussian noise (with varying standard deviation) was added to the coordinates of the image points–the results are presented in the right-hand plot of Figure 5.6.

Figure 5.7: Inextensible surface: Real deformations. Courtesy of [33]

Comparative Study: In addition to the evaluation of my procedure, I compare my approach with other approaches in the literature. The approaches considered were:

Method 1: Inextensible - the SOCP-based algorithm proposed in [32], in which the authors use a triangular mesh together with a-priori datasets for learning the deformations. The deformation constraints are expressed using the upper-bound model.

Method 2: Inextensible - The approach of [56] where the authors used differential models to represent the deformation constraints and applied an optimization scheme based on these models.

Method 3: Extensible - The work of [51] in which the authors formulate the reconstruction problem of a generic surface in terms of the minimization of stretching energy and impose a set of fixed boundary 3D points to constrain the solution.

As presented in the left-hand chart in Figure 5.5, the methods 1 and 2 are affected by significant errors in the case of extensible deformation, whereas my method performs satisfactorily for both deformation types. The right-hand chart in Figure 5.5 shows that results obtained with method 3 display significant errors in the case of inextensible deformation even though results for extensible deformatins have small errors.

Real Data: I also applied my reconstruction procedure to images of real deformations acquired by a camera. The datasets contain 2D image points and their corresponding posi-

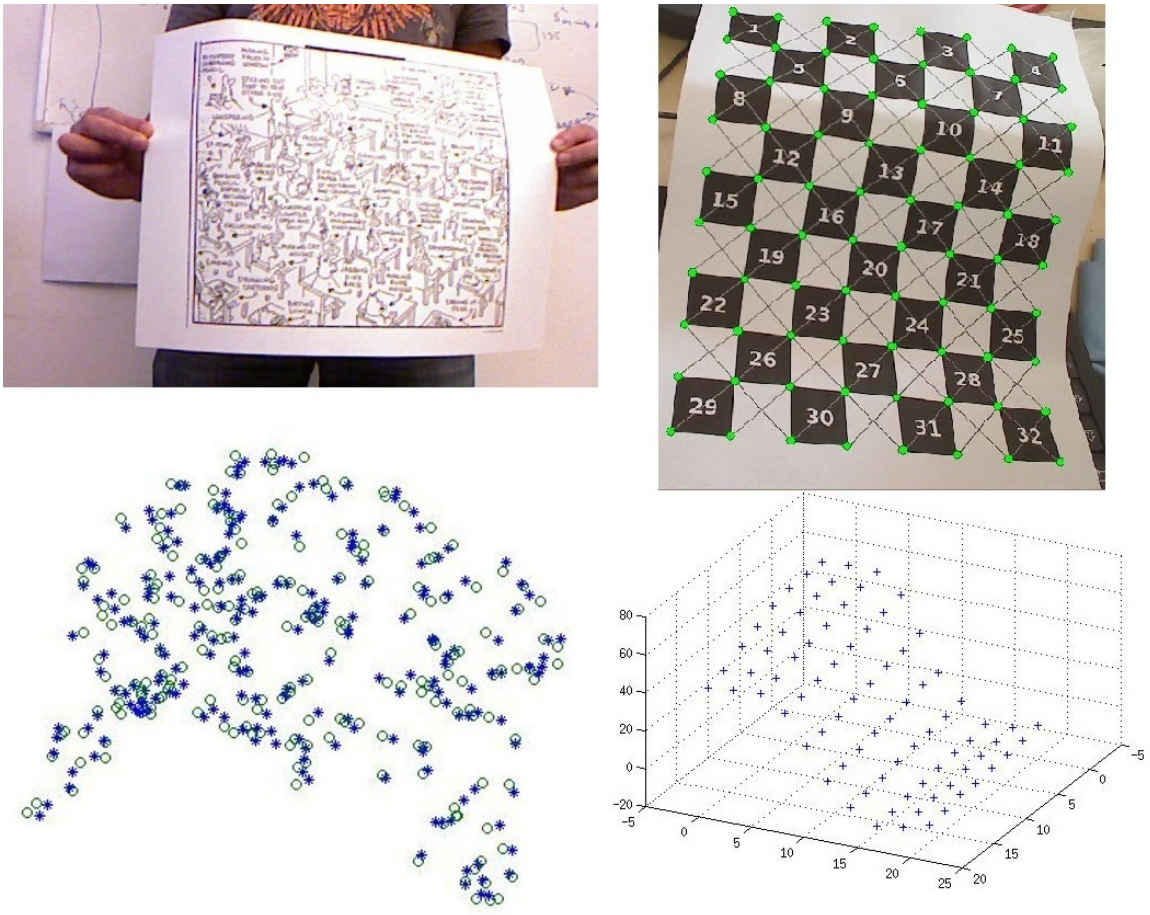

Figure 5.8: Top: Sample deformations (Courtesy of [33, 56]). Bottom: The reconstructed points of the deformations above. The points marked with '*' and those marked with 'o' show the estimated and the ground-truth points on the paper-sheet, respectively.

tions in the reference image (undeformed shape). The test objects include a paper-sheet, a cushion, and a piece of fabric, etc for the inextensible case, as shown in Figures 5.7. Figure 5.8 shows 2 deformations and the corresponding reconstructed shapes. In this Figure, the left-hand surface has an irregular point distribution, whereas the right-hand surface is represented by points in a fixed pattern. Also, I use beach and football balls for the case of extensible deformations, as shown in Figure 5.9. In this case, the desired deformations are obtained by inflating/deflating these objects and 8 points per patch are considered. For extensible deformations I use a random shape as the reference shape. The 3D positions of the points have been obtained with acceptable accuracy.

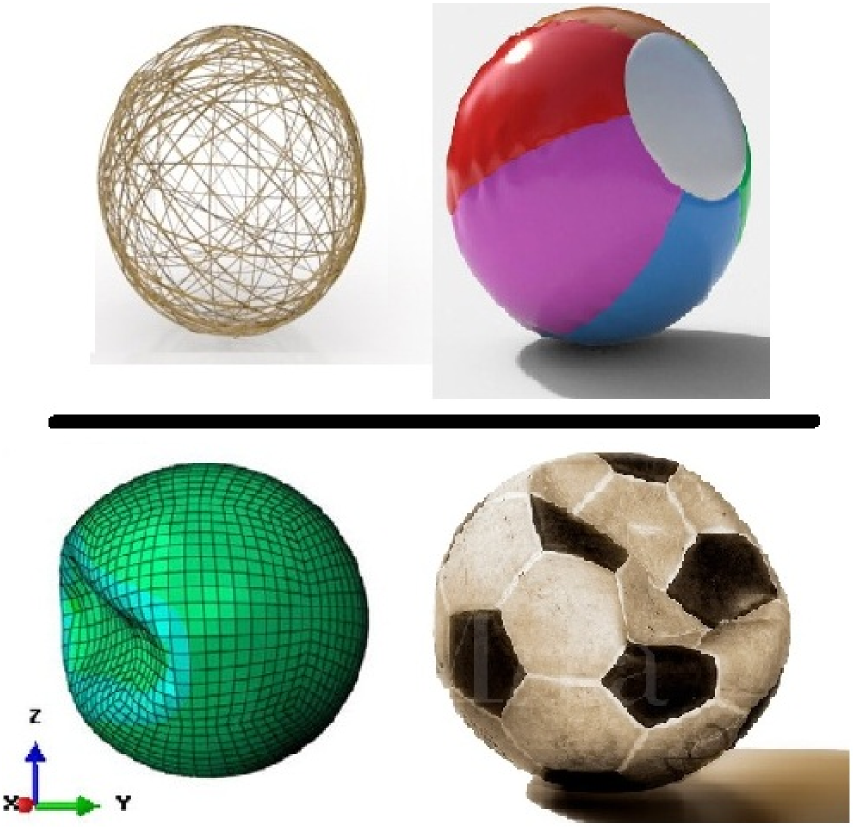

Figure 5.9: Top: Extensible deformations using beach and football balls.

5.6 Conclusions

We presented an approach to reconstruct deformable surfaces using a single image. The deformations can range from inextensible to extensible, with the latter undergoing expansion/contraction. To perform this reconstruction, a reference 3D shape was used. On the other hand, I employed the observation that a real surface usually displays local homogeneous deformations which together create a more general deformations of the surface. To compute the deformartion and perform the 3D reconstruction, the surface was divided into patches. Given the undeformed shape, this was formulated as an optimization scheme, intended to estimate the 3D positions of the deformed points from one image. The optimization acts on each patch separately. The experimental results show that this approach can perform 3D reconstructions of several types of deformations with acceptable precision.

Reconstructions Based on Homothetic Deformation

5.7 Problem Statement

In this subtask, I tackle the problem of reconstructing extensible surfaces, specially those that may extend or be enlarged in volume and take the form of volumetric shapes similar to the sphere, cylinder and the ellipsoid. Examples are plastic balloons, hearts, balls - see Figure 5.10. Such objects undergo a continuous, relatively uniform expansion over any small patch of the surface. This expansion usually occurs along a certain axis. Given a 3D template consisting of known 3D points and its image, the objective is to determine the 3D positions of a set of points in any image, from a video sequence acquired with a calibrated camera. Correspondences are established with their positions in the template image. In my formulation of the reconstruction problem, I assume that the whole surface deformation is locally homothetic i.e. homothetic deformation implies uniform expansion. As a result, the surface is treated as being built from patches connected together, with the deformation of each patch being considered homothetic. This kind of deformation can be represented by a linear function that transforms the 3D template into a deformed patch. This function is decoupled into a pure deformation and a rigid body transformation. My approach is developed in such a way that we start by estimating the pure deformation first. Therefore an optimization procedure is defined so that only the pure deformation is estimated. The rigid body transformation can be then estimated. This is done by decomposing the essential matrix between the current image and the virtual image that corresponds to the projection of the deformed points following the application of the pure deformation. Finally, as the result of the estimation of the linear mapping, we can compute the 3D positions of the points for each surface patch, separately, up to a scaling factor. Each patch is assumed to overlap only up to two neighboring patches. That assumption allows a smooth reconstruction up to a global scale, by enforcing the smoothness of the surface.

Figure 5.10: Test objects that can be expanded by inflation.

Organization of this subtask: First, I describe the decomposition of the linear mapping. The optimization itself is explained in detail in the following section. Next. the experimental results are presented, followed by conclusions.

5.8 Proposed Approach: Deformation Model

As disscussed earlier, the deformation gradient **F** is expressed as:

$$\mathbf{F} = \begin{bmatrix} \frac{\delta x_i}{\delta x_i^{\circ}} & \frac{\delta x_i}{\delta y_i^{\circ}} & \frac{\delta x_i}{\delta z_i^{\circ}} \\ \frac{\delta y_i}{\delta x_i^{\circ}} & \frac{\delta y_i}{\delta y_i^{\circ}} & \frac{\delta y_i}{\delta z_i^{\circ}} \\ \frac{\delta z_i}{\delta x_i^{\circ}} & \frac{\delta z_i}{\delta y_i^{\circ}} & \frac{\delta z_i}{\delta z_i^{\circ}} \end{bmatrix} \tag{5.9}$$

F can be written as either **R**.**U** or **V**.**R** through a polar decomposition. In each case, **R** is the rotation matrix, and **U** and **V** are symmetric matrices describing stress and strain, which contribute to deformations.

Uniformly extensible deformations are homothetic. These deformations are characterized by having the same gradient at every point, with the gradient being a special case of the gradient for homogeneous deformation. As a result, the mapping from the undeformed shape, which is chosen as the template, onto the deformed surface can be formulated as:

$$\mathbf{p}_i = \mathbf{F}.\mathbf{p}_i^{\circ} + \mathbf{c} \tag{5.10}$$

where $\mathbf{c}$ is a 3×1 vector that denotes the rigid body translation. We refer to this linear mapping as deformation model. Every surface patch is then represented separately by means of this mapping. Let us denote $\mathbf{F}$ as $\mathbf{R}.\mathbf{U}$. I propose an algorithm for calculating the rigid body transformation depending on the deformation matrix $\mathbf{U}$ to be given in advance. The algorithm below shows how to perform the calculation:

For a surface patch, the 3D positions that result from the equation $\hat{\mathbf{p}}_i = \mathbf{U}.\mathbf{p}_i^{\circ}$ are on a virtual patch that can be viewed as intermediate between the template and the real deformed patch. The corresponding points on both the virtual and the real patch are related as follows:

$$\mathbf{p}_i = \mathbf{R}.\hat{\mathbf{p}}_i + c \tag{5.11}$$

This equation basically defines a transformation between 2 coordinate systems. However, in our case, $\mathbf{R}$ and c specifies a transformation between $\hat{\mathbf{p}}_i$ and $\mathbf{p}_i$ in the same coordinate system (i.e. camera coordinates). To deal with this transformation, we consider it differently, in the sense that, with no loss of generality, we can assume that $\mathbf{R}$ and c indicate the transformation between the current image and a virtual image - see Figure 5.11. This is a dual configuration of the real problem under consideration. The virtual image, in fact, represents only 2D image points obtained by projecting the intermediate patch onto the image plane with the calibration matrix. As a result, we now have one rigid surface patch projected onto two separate images that are related by $\mathbf{R}$ and c. By decomposing the essential matrix, $\mathbf{R}$ and c are then estimated. To do so, the following procedure can be followed:

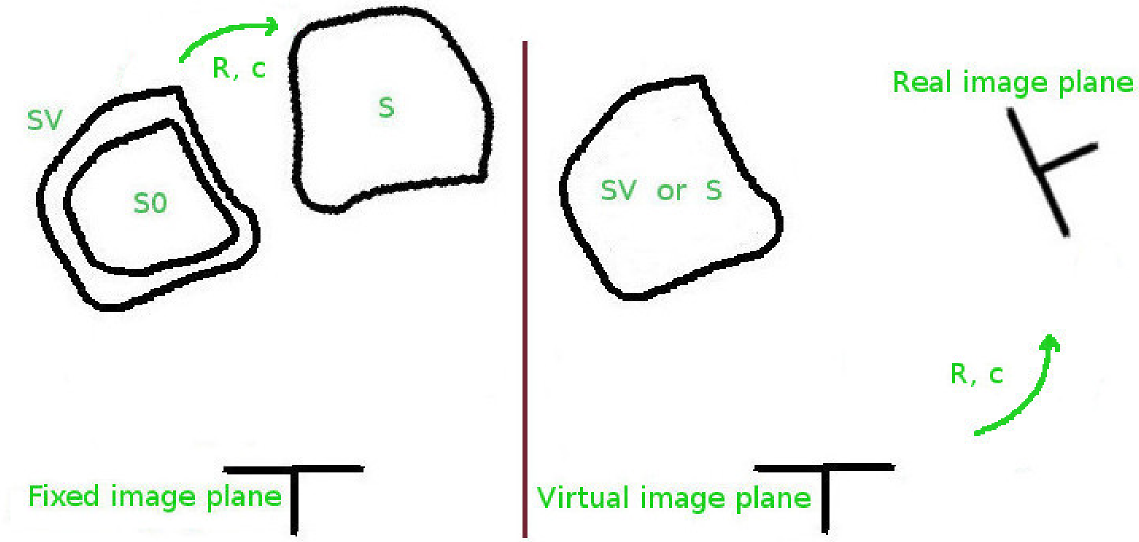

Figure 5.11: Left-side image: the real problem. Right-side image: the dual problem. **S0**, **SV** and **S** indicate the template, the intermediate patch and the real deformed patch respectively.

1-Estimate the essential matrix between the two images by using the corresponding image points for the patch.

2-Decompose the essential matrix into rotation and translation.

3- This decomposition yields 4 possible solutions [65]. To find the one that is feasible in our problem, the positive depth constraint can be applied.

Having estimated **R** and c, the deformed surface can be reconstructed up to scale. However, the algorithm just described depends on our knowledge of the deformation matrix U. Since this matrix is unknown, we have to determine it. For this purpose, we formulate an optimization procedure in which **U** is the unknown. Such optimization requires an initial estimate of **U**. The 3-step algorithm above forms the core of my optimization procedure. For simplicity's sake, I call it *pose estimation algorithm*.

5.9 Proposed Approach: Optimization

Consider that **K** is a known calibration matrix with focal length f and that $\mathbf{q} = \begin{pmatrix} u, & v \end{pmatrix}$ denotes the image points - corresponding to **p** - projected in the image of the deformed surface. For point i, the projection equations are defined by $f.p_{x,i} - u_i.p_{z,i} = 0$ and $f.p_{y,i} - v_i.p_{z,i} = 0$. These equations are used to define the reprojection error to be minimized. I also define a second constraint, intended to ensure that the deformation is approximately homothetic. This is equivalent to saying that the off-diagonal entries of **U** should be approximately zero. This results from the fact that the diagonal entries of the deformation matrix **U** are dominant for a homothethic surface, compared with the off-diagonal ones. The optimization procedure includes the pose estimation algorithm, reprojection error and the constraint for homothetic deformation. Algorithm 1 details the different steps of the optimization procedure.

Algorithm 3: Optimization procedure to estimate $\mathbf{U}_0$.

```
Initialization of $\mathbf{U}_0$.
while the total error is not lower than a specified threshold do
  Update $\mathbf{U}_0$.
  $\begin{bmatrix} \mathbf{R}_0, & \mathbf{U} \end{bmatrix} = \mathrm{PolarDecom}(\mathbf{U}_0)$
  Pose estimation with U as input.
  Minimization of the total error including reprojection error and the other constraint error.
end while
The iteration is stopped and the current value for $\mathbf{U}_0$ is used.
```

Next I describe this algorithm in detail:

Step 1: $\mathbf{U}_0$ is a 3×3 matrix that is to be estimated as the result of the optimization. This matrix is initialized with random values.

Step 2: $\mathbf{U}_0$ is factorized by a polar decomposition into a deformation matrix **U**, which is symmetric, positive semi-definite and a rotation matrix $\mathbf{R}_0$ which is not used.

Step 3: **U**, is used in the pose estimation algorithm. At this point, **R** and c are determined.

Step 4: After having estimated all the parameters of the deformation model, approximate 3D positions can be computed using Equation 5.10. Then, the following error is minimized with respect to $\mathbf{U}_0$:

$$\min_{\mathbf{U}_0} \sum_{i=1}^{n} \left[\left(f.p_{x,i} - u_i.p_{z,i} \right)^2 + \left(f.p_{y,i} - v_i.p_{z,i} \right)^2 \right] + w.\sum \text{OffDiagonalEntries}\left(\mathbf{U}\right)^2 \tag{5.12}$$

where the latter term indicates the sum of squares of off-diagonal entries of $\mathbf{U}$ and w is a weighting factor which is set empirically. n is the number of points.

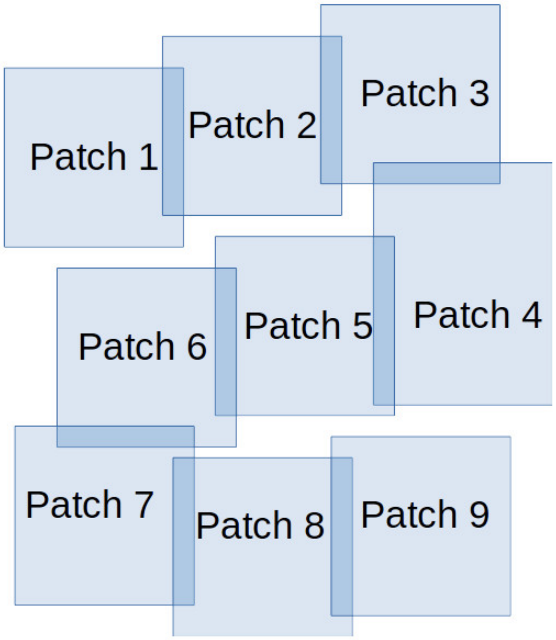

Figure 5.12: The chain-like form of patches

The above optimization is a non-linear sum of squares problem that can be solved using the Levenberg-Marquardt algorithm.

Step 5: If the step error (that is, the size of the change in the location where the objective function was evaluated) is not lower than a specified threshold, the optimization is iterated starting with a new random value for $\mathbf{U}_0$.

Step 6: Once the value for matrix $\mathbf{U}_0$ is estimated, we can solve for all the parameters of the deformation model by using steps 2 and 3. As a result, the 3D positions of the points are determined up to a scaling factor. This factor is due to the fact that translation c is estimated only up to a scale factor. The whole approach proposed so far is applied separately to all surface patches. Consequently, each will be reconstructed on a possibly different scale. In order to obtain a common scale, surface smoothness is enforced by considering the points where the patches overlap. Assume that the patches are labeled and that there are a total of n_a patches. Every 2 successive patches overlap in chain-like form, as shown in Figure 5.12. The smoothness enforcement is accomplished with this formula:

$$scale = \frac{1}{n_c} \sum_{i=1}^{n_c} \left(\frac{1}{3} \sum_{j=1}^{3} p_{i,j,l} / p_{i,j,k} \right) \tag{5.13}$$

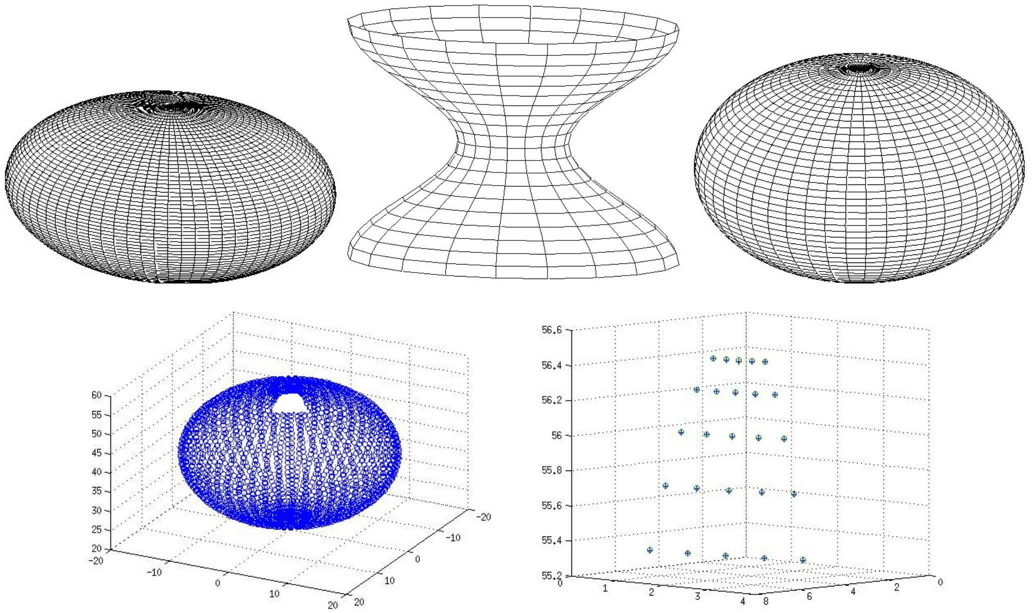

Figure 5.13: Synthetic extensible surfaces.Top: from left to right: ellipsoid, deformed cylinder, sphere. Bottom: Left image shows a patch of the sphere, while the Right image shows the points on the surface. The estimated and the ground-truth points are marked with '+' and 'o', respectively.

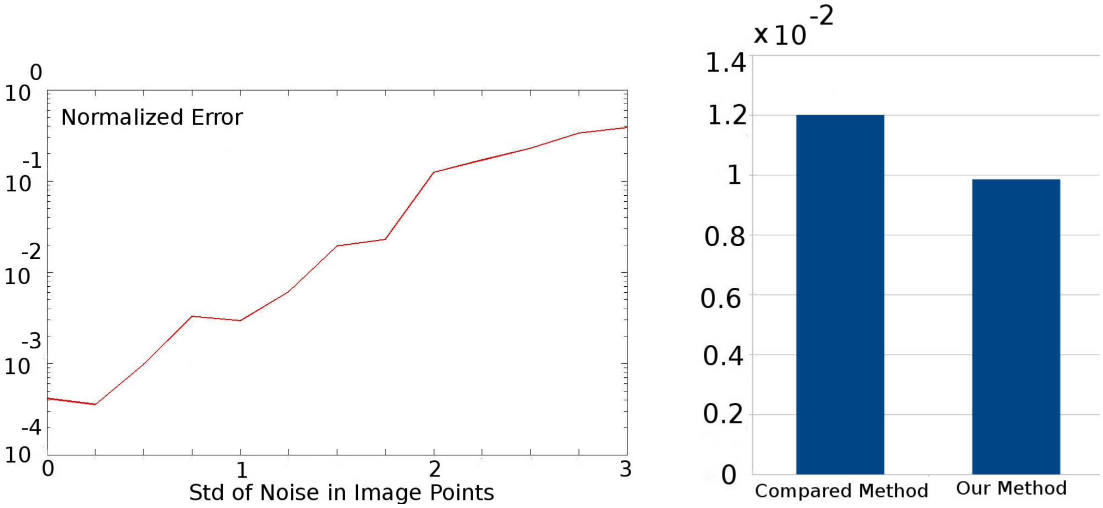

Figure 5.14: Left: Reconstruction error with respect to noise added to image points. Right: A comparison of reconstruction error between my approach and the approach of [51].

where n_c is the number of common points belonging to patches l and k. Subscript j denotes the x, y and z coordinates. Patch 1 is selected to be the scaling reference and all the patches will be rescaled with respect to it. Equation 5.13 is employed for every 2 overlapping patches, sequentially, up to patch n_a.

5.10 Experimental Results

In this section, I validate the proposed approach by conducting a set of experiments on data that conforms to the type of deformation I am addressing. For that purpose I consider simple examples, namely the cylinder, ellipsoid and the sphere- see Figure 5.13. Their mathematical models can be used to generate the set of n point correspondences required. For example, the sphere is expanded/contracted uniformly by changing its radius. Although these volumetric shapes are simple surfaces, they constitute adequate models for the tests with synthetic data, so that the efficiency of the approach can be evaluated quantitatively. Points $\mathbf{p}$ are assumed to be visible from the virtual camera, as illustrated in Figure 5.13. There are a total of 185 points which are divided into 10 patches, each of which has the same number of points. This is done by considering 5 common points between any two overlapping patches and so we have 25 points per patch. The accuracy of the approach is evaluated using the 3D reconstruction error which is defined by:

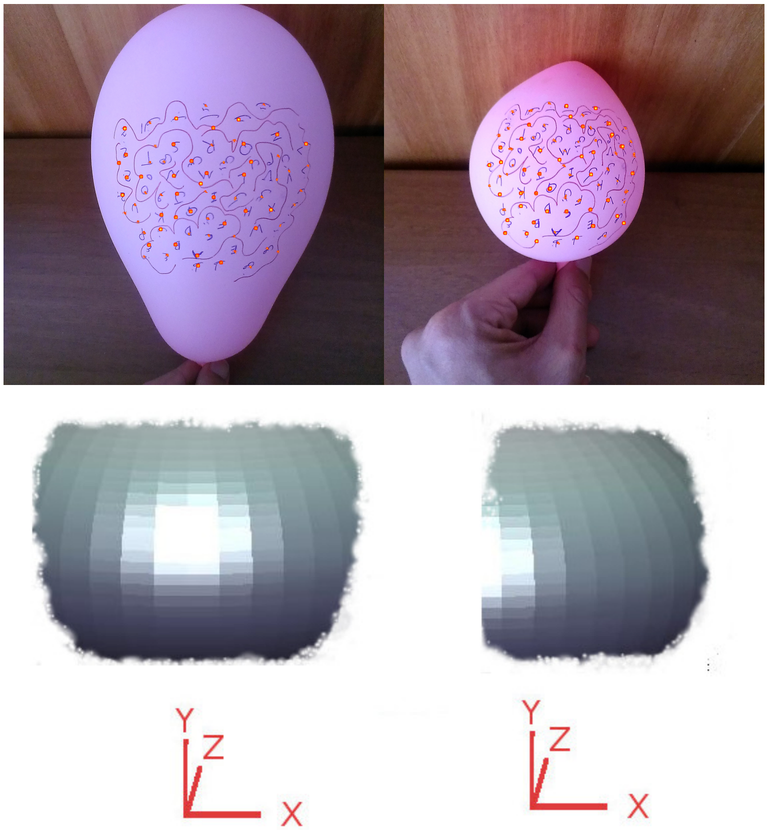

Figure 5.15: Top: Real extensible deformations using plastic baloons with the reconstructed points indicated by dots. Bottom: The surface fit to the points.

$$RE = \frac{1}{n}\sum_{i=1}^{n}\left[\|\mathbf{p}_i - \hat{\mathbf{p}}_i\|^2 \,/\, \|\hat{\mathbf{p}}_i\|^2\right] \tag{5.14}$$

This error corresponds to the normalized Euclidean distance between the observed ($\hat{\mathbf{p}}_i$) and the estimated ($\mathbf{p}_i$) world points where n is the total number of points. Taking into consideration that the reconstruction estimates the 3D coordinates up to scale with respect to the ground-truth data, an estimate of the scaling factor is used in the calculation of the reconstruction error. In addition to the case of synthetic data without noise, I also estimate the reconstruction error by adding noise to the coordinates of the 2D image points (Fig. 5).

The results shown in the left-side plot correspond to average values obtained by repeating the estimation 100 times for each one of 15 different random deformations used. Five different random deformations were considered for each synthetic surface. As expected, the number of iterations required for the optimization to converge increases with noise. In the absence of noise the average number of iterations required for convergence is 150.

The experiments with real images were performed using a plastic baloon. The desired deformations are obtained by inflating this object. 3D ground-truth data corresponding to a template shape is also computed- see Figure 5.15. The image of the deformed surface is acquired by a calibrated camera. The surface includes a total of 80 points which are grouped into 7 patches with 3 points on each overlapping area. Each patch has 14 points. The input data contains 2D point correspondences matched to the template image.

I compare the performance of the proposed approach with the approach proposed in [51], where the authors formulate the reconstruction of a generic surface in terms of the minimization of stretching energy while imposing a set of fixed boundary 3D points to constrain the solution - see Figure 5.14. The results given on the right-side plot show the averages obtained by repeating the estimation 100 times for each of 10 deformations of the plastic baloon.

5.11 Conclusions

In this subtask, I have proposed a reconstruction method for elastic surfaces that can expand freely. This approach is based on the assumption that the deformation is locally homothetic (i.e. uniformly extensible), and reconstructs the surface by splitting it into overlapping patches. Each local deformation was then modeled by a linear mapping between the image and a known 3D template. The mapping is defined by a combination of a pure deformation and a rigid-body transformation. The deformation and rigid transformation were estimated by means of an optimization procedure. As a result, estimates of the 3D coordinates (up to scale) of the points on each patch are obtained. A global scale factor for all the patches is obtained by enforcing the smoothness of the reconstruction. The results demonstrated the efficiency of the approach.

CHAPTER 6

CONCLUSIONS AND DISCUSSIONS

6.1 Thesis Summary

In this dissertation, we took into full consideration the problem of reconstructing surfaces that deform in a natural way. The deformation could take two general forms, namely inextensible and extensible. Each of these deformation types has been carefully studied by reviewing key concepts to sophisticated theories and then efficient approaches have been proposed accordingly after going through elementary solutions to advanced methodologies.

The whole work carried out was an organized effort to accomplish the thesis objectives in a 3-stage research program. Each stage was allocated to one task and we dealt with the reconstruction problem in various ways. To be precise, in each task certain algorithms have been designed in accordance with the elements defined in the reconstruction problem. Therefore, three general ideas have been proposed in order to perform the surface reconstruction. Below, we discuss these ideas and compare them with respect to advantages and disadvantages.

6.2 Depth Camera Combined with a Monocular Camera

Strength: a ToF camera capable of supplying relatively accurate depth measurements can be regarded as an auxiliary tool to assist 3D technology in creating highly precise applications, specially when the need for prior data or knowledge is essential to the performance of a reconstruction system. Likewise, because having prior learning data is an indispensable requirement for surface reconstructions based on a mesh representation, we employed a ToF camera calibrated relative to an RGB camera in order to acquire this data. This is an evident benefit of depth cameras.

Summary of the first task: we dealt with reconstruction of isometric surfaces. To perform such monocular reconstruction, an algorithm based on the linear deformation model and consisting of a non-linear least squares optimization was proposed. To find the proper deformation model, prior training data should be used. We therefore provided this prior data by proposing a novel approach for the reconstruction of a typical surface so that the computed deformation model can be also extended to other isometric surfaces. This approach was founded on a range camera along with a conventional camera and its goal is to estimate the 3D positions of the mesh vertices from the depth of the feature points. By applying this approach to multiple mesh deformations we acquired the training data required. Experimental results showed that both the proposed reconstruction schemes are efficient and result in accurate reconstructions.

Weakness: the results achieved with this camera configuration were found to be excellent in comparison with those obtained from the other two ideas. However, since a system equipped with a depth camera is costly and also limited to particular operating situations, the necessity of such cameras should be justified when an equally efficient system exists that only relies on a conventional camera. This implies that minimal hardware in cases like this is an issue to consider. Consequently, the two approaches discussed next makes use of a monocular camera.

6.3 Joint Use of Deformation Constraints

Strength: in the absence of depth cameras, surface reconstruction becomes more complicated. In order to reduce that complexity, an obvious idea is to highly constrain the problem with the help of physical requirements imposed by the type of deformation (i.e. deformation constraints). As a result, the two sets of deformation constraints already introduced in the context (upper-bound model and differential model) have been reformulated in a way that facilitates the joint use of these, thus leading to a well-constrained problem, solved by means of a semi-definite programming problem for the reconstruction of inextensible surfaces. The results of this approach were shown to be comparable to those of the previous approach. Although it is not able to perform as effectively, however this one offers the advantage of using only a conventional camera over the other. Also, in line with this idea, a second-order cone programming problem has been defined for the reconstruction of conformal surfaces separately.

Summary of the second task: this task was made up of two subtasks. In the first subtask, monocular reconstruction of deformable, inextensible surfaces using one image was addressed. To perform this reconstruction, an SDP optimization was formulated by imposing constraints corresponding to the type of deformation i.e. deformation constraints. These constraints were defined using a differential model of the deformation, and also the upper-bound model. The application of the differential model requires that a parametric mapping between the template and the 3D surface be defined. We used a Free-Form Deformation for this purpose. Both types of constraints were then combined in a SDP optimization program. The optimization estimated the 3D positions of the points on the deformed surface. The experimental results show and characterize the performance of the proposed approach. The second subtask was an investigation of conformal deformation reconstruction, aiming to propose an easy-to-implement methodology that still attains accurate reconstructions. The results show that the approach is reasonably efficient. Assuming that the boundaries/contours of the surface undergo negligible motion, a set of estimates was obtained for the 3D positions of points on the surface using a modified, variational formulation of the upper-bound model and The best estimate was then selected based on some criteria.

Weakness: a primary issue with a reconstruction algorithm based on deformation constraints is that it only performs well when applied to deformations classified as having a particular type of deformation. Therefore, such an approach fails to give accurate results if the deformation deviates from the deformation type specified in the problem, as the deformation constraints will not be satisfied as expected.

6.4 Deformation Model Based on Surface Transformations

Strength: deformation constraints are specially necessary when a specific deformation type is to be reconstructed, meaning that these are not applicable if the surface undergoes deformation with unknown degrees of freedom. In this case, surface transformations provide a convenient way to model the deformation and allow for the reconstruction of surfaces showing a broad class of deformations. On the assumption that a natural surface displays local homogeneous deformations, we proposed a patch-based algorithm based on a linear deformation model in order to reconstruct surfaces regardless of their deformation types. This approach is point-wise and gives satisfactory results.

Summary of the third task: this task also contained two subtasks. In the first subtask, we presented an approach to reconstruct deformable surfaces using a single image. The deformations can range from inextensible to extensible, with the latter undergoing expansion/contraction. To perform this reconstruction, a reference 3D shape was used. On the other hand, we employed the observation that a real surface usually displays local homogeneous deformations which together create a more general deformations of the surface. To compute the deformartion and perform the 3D reconstruction, the surface was divided into patches. Given the undeformed shape, this was formulated as an optimization scheme, intended to estimate the 3D positions of the deformed points from one image. The optimization acts on each patch separately. The experimental results show that this approach can perform 3D reconstructions of several types of deformations with acceptable precision. In the second subtask, we have proposed a reconstruction method for elastic surfaces that can expand freely. This approach is based on the assumption that the deformation is locally homothetic (i.e. uniformly extensible), and reconstructs the surface by splitting it into overlapping patches. Each local deformation was then modeled by a linear mapping between the image and a known 3D template. The mapping is defined by a combination of a pure deformation and a rigid-body transformation. The deformation and rigid transformation were estimated by means of an optimization procedure. As a result, estimates of the 3D coordinates (up to scale) of the points on each patch are obtained. A global scale factor for all the patches is obtained by enforcing the smoothness of the reconstruction. The results demonstrated the efficiency of the approach.

Weakness: Compared with the other two approaches, this approach shows a lower level of accuracy. However, that deficiency is compensated at the cost of a more general reconstruction scheme.

The research conducted for this thesis is a starting point for more remarkable contributions in the future.

Appendices

APPENDIX A

MATHEMATICAL NOTATIONS

Throughout this thesis, we use the convention listed below as mathematical symbols.

Matrices are represented as bold capital letters ($\mathbf{A} \in \mathbb{R}^{n \times m}$, n rows and m columns). Vectors are represented as bold lower-case letters ($\mathbf{a} \in \mathbb{R}^n$, n elements). By default, a vector is considered a column. Lower-case letters (a) represent one dimensional elements. By default, the jth column vector of $\mathbf{A}$ is specified as $\mathbf{a}_j$. The jth element of a vector $\mathbf{a}$ is written as a_j. The element of $\mathbf{A}$ in the row i and column j is represented as $a_{\mathrm{i,j}}$. $\mathbf{A}^{(1:2)}$ indicates the first 2 rows of $\mathbf{A}$. $\mathbf{A}^{(3)}$ denotes the third row of $\mathbf{A}$. Regular capital letters (A) indicate one dimensional constants. We use $\mathbb{R}$ after a vector or matrix to denote that it is represented up to a scale factor.

CHAPTER 7

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] M. Paladini, A. Bue, M. Stosic, M. Dodig, J. Xavier, and L. Agapito, “Factorization for nonrigid and articulated structure using metric projections.,” Proc. IEEE Conf. on Computer Vision and Pattern Recognition, 2009, p. 2898.2905.

[2] Y. Dai, H. Li, and M. He, “A simple prior free method for nonrigid structure from motion factorization,” CVPR, 2012, p. 2018.2025.

[3] D. Terzopoulos, J. Platt, A. Barr, and K. Fleicher, “Elastically deformable models,” ACM SIGGRAPH, 1987, pp. 205–214.

[4] C. Nastar and N. Ayache, “Frequency.based nonrigid motion analysis,” PAMI, 18(11): 1067.1079, 1996.

[5] A. Pentland and S. Sclaroff, “Closed.form solutions for physically based shape modeling and recognition,” PAMI, 13(7):715.729, 1991.

[6] A. Agudo and F. Moreno-Noguer, “Simultaneous pose and non-rigid shape with particle dynamics,” IEEE Conference on Computer Vision and Pattern Recognition, 2015, p. 2179.2187.

[7] J. Xiao and T. Kanade, “Uncalibrated perspective reconstruction of deformable structures,” ICCV, 2005, pp. 1075–1082.

[8] M. Brand, “Morphable 3d models from video,” CVPR, 2001, p. 456.463.

[9] A. Bartoli and S. Olsen, “A batch algorithm for implicit nonrigid shape and motion recovery,” In ICCV Workshop on Dynamical Vision, 2005.

[10] I. Akhter, Y. Sheikh, and S. Khan, “In defense of orthonormality constraints for nonrigid structure from motion,” CVPR, 2009, p. 1534.1541.

[11] X. Llado, A. DelBue, and L. Agapito, “Nonrigid 3d factorization for projective reconstruction,” BMVC, vol. 28, 2005.

[12] J. Xiao, J. Chai, and T. Kanade, “A closed.form solution to nonrigid shape and motion recovery,” ECCV, 2004, p. 573.587.

[13] H. Zhou, X. Li, and A. Sadka, “Nonrigid structure from motion from 2.d images using markov chain monte carlo,” MultMed, vol. 14, no. 1, p. 168.177, 2012.

[14] A. Agudo, F. Moreno-Noguer, B. Calvo, and J. Montiel, "Sequential non-rigid structure from motion using physical priors," Pattern Analysis and Machine Intelligence, p. 679.994, 2015.

[15] H. Aanaes and F. Kahl, "Estimation of deformable structure and motion," Workshop on Vision and Modelling of Dynamic Scenes, ECCV, Denmark, 2002.

[16] A. DelBue, X. Llado, and L. Agapito, "Nonrigid metric shape and motion recovery from uncalibrated images using priors," IEEE Conference on Computer Vision and Pattern Recognition, New York, vol. 1, 2006, p. 1191.1198.

[17] C. Tomasi and T. Kanade, "Shape and motion from image streams under orthography: A factorization approach," International Journal of Computer Vision, 9(2): 137.154, 1992.

[18] C. Bregler, A. Hertzmann, and H. Biermann, "Recovering nonrigid 3d shape from image streams," IEEE Conference on Computer Vision and Pattern Recognition, 2000, p. 2690.2696.

[19] L. Torresani, D. Yang, E. Alexander, and C. Bregler, "Tracking and modeling nonrigid objects with rank constraints," IEEE Conference on Computer Vision and Pattern Recognition, 2001.

[20] A. Bartoli, V. GayBellile, U. Castellani, J. Peyras, S. Olsen, and P. Sayd, "Coarse to fine low rank structure from motion," IEEE Conference on Computer Vision and Pattern Recognition, 2008.

[21] L. Torresani, A. Hertzmann, and C. Bregler, "Nonrigid structure from motion: Estimating shape and motion with hierarchical priors," PAMI, 30(5):878.892, 2008.

[22] M. Brand, "A direct method for 3d factorization of nonrigid motion observed in 2d," IEEE Conference on Computer Vision and Pattern Recognition, 2005, p. 122.128.

[23] T. Akhter, Y. Sheikh, S. Khan, and T. Kanade, "Trajectory space: A dual representation for nonrigid structure from motion," International Journal of Computer Vision, vol. 33, p. 1442.1456, 2011.

[24] X. Llado and et.al, "Reconstruction of non-rigid 3d shapes from stereo-motion," Pattern Recognition Letters, p. 1020.1028, 2011.

[25] R. Hartley and R. Vidal, "Perspective nonrigid shape and motion recovery," European Conference on Computer Vision, 2008, p. 276.289.

[26] C. Russell, R. Yu, and L. Agapito, "Video pop-up: Monocular 3d reconstruction of dynamic scenes," ECCV, 2014, p. 583.598.

[27] J. Costeira and T. Kanade, “A multibody factorization method for independently moving objects,” International Journal of Computer Vision, p. 159.179, 1998.

[28] K. Kanatani, “Motion segmentation by subspace separation: Model selection and reliability evaluation,” International Journal of Image and Graphics, p. 179.197, 2002.

[29] A. DelBue, X. Llado, and L. Agapito, “Segmentation of rigid motion from nonrigid 2d trajectories,” The 3rd Iberian conference on Pattern Recognition and Image Analysis, 2007, p. 491.498.

[30] A. DelBue, F. Smeraldi, and L. Agapito, “Non-rigid structure from motion using non-parametric tracking and non-linear optimization,” Computer Vision and Pattern Recognition Workshop, 2004, p. 491.498.

[31] V. GayBellile, M. Perriollat, A. Bartoli, and P. Sayd, “Image registration by combining thin.plate splines with a 3d morphable model,” International Conference on Image Processing, 2006, p. 1069.1072.

[32] M. Salzmann, R. Hartley, and P. Fua, “Convex optimization for deformable surface 3.d tracking,” IEEE International Conference on Computer Vision, 2007, p. 1.8.

[33] M. Salzmann and P. Fua, “Reconstructing sharply folding surfaces: A convex formulation,” IEEE Conference on Computer Vision and Pattern Recognition, 2007, p. 1054.1061.

[34] N. Gumerov, A. Zandifar, R. Duraiswami, and L. Davis, “Structure of applicable surfaces from single views,” European Conference on Computer Vision, vol. 3023, 2004.

[35] M. Prasad, A. Zisserman, and A. Fitzgibbon, “Single view reconstruction of curved surfaces,” IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, 2006, p. 1345.1354.

[36] F. Bookstein, “Principal warps: Thin.plate splines and the decomposition of deformations,” PAMI, 11(6):567.585, 1989.

[37] M. Salzmann, R. Urtasun, and P. Fua, “Local deformation models for monocular 3d shape recovery,” IEEE Conference on Computer Vision and Pattern Recognition, 2008, p. 1.8.

[38] S. Shen, W. Shi, and Y. Liu, “Monocular 3.d tracking of inextensible deformable surfaces under l2.norm,” IEEE Transactions on Image Processing 19, vol. 19, p. 512.521, 2010.

[39] M. Perriollat, R. Hartley, and A. Bartoli, “Monocular template.based reconstruction of inextensible surfaces,” International Journal of Computer Vision, 2010.

[40] M. Salzmann, F. MorenoNoguer, V. Lepetit, and P. Fua, “Closed.form solution to nonrigid 3d surface registration,” European Conference on Computer Vision, 2008, p. 581.594.

[41] S. Parashar, D. Pizarro, and A. Bartoli, “Isometric non-rigid shape-from-motion in linear time,” CVPR, 2016.

[42] L. Cohen and I. Cohen., “Finite.element methods for active contour models and balloons for 2.d and 3.d images,” PAMI, 15(11):1131.1147, 1993.

[43] D. Metaxas and D. Terzopoulos, “Constrained deformable superquadrics and nonrigid motion tracking,” PAMI, vol. 15, p. 580.591, 1993.

[44] R. White and D. Forsyth, “Combining cues: Shape from shading and texture,” CVPR, 2006.

[45] M. Gallardo, T. Collins, and A. Bartoli, “Using shading and a 3d template to reconstruct complex surface deformations,” BMVC, 2016.

[46] A. Chhatkuli, D. Pizarro, and A. Bartoli, “Inextensible non-rigid shape-from-motion by second-order cone programming,” CVPR, 2016.

[47] A. Bartoli, Y. Gérard, F. Chadebecq, and T. Collins, “On template.based reconstruction from a single view: Analytical solutions and proofs of well.posedness for developable, isometric and conformal surfaces,” CVPR, 2012, p. 2026.2033.

[48] F. Brunet, R. Hartley, A. Bartoli, N. Navab, and R. Malgouyres, “Montiel. finite element based sequential bayesian nonrigid structure from motion,” CVPR, 2012.

[49] A. Chhatkuli, D. Pizarro, and A. Bartoli, “A stable analytical framework for isometric shape-from-template by surface integration,” 2016.

[50] A. Agudo, B. Calvo, and J. Montiel, “Fem models to code nonrigid ekf monocular slam.,” IEEE Workshop on Dynamic Shape Capture and Analysis of ICCV, 2011.

[51] A. Malti, R. Hartley, A. Bartoli, and T. Collins, “Monocular template.based 3d reconstruction of extensible surfaces with local linear elasticity,” CVPR, 2013.

[52] J. Diebel and S. Thrun, “An application of markov random fields to range sensing,” Proc. NIPS, 2005, p. 291.298.

[53] Y. Kim, C. Theobalt, J. Diebel, J. Kosecka, B. Miscusik, and S. Thrun, “Multi view image and tof sensor fusion for dense 3d reconstruction,” Computer Vision Workshops (ICCV Workshops), 2009, p. 1542.1549.

[54] R. Yang, J. Davis, and D. Nister, “Spatial.depth super resolution for range images,” Computer Vision and Pattern Recognition, CVPR ’07. IEEE Conference, Minneapolis, MN, 2007, p. 1.8.

[55] H. Kim, Y. Tai, and M. Brown, “High quality depth map upsampling for 3d.tof cameras,” Inso Kweon Computer Vision (ICCV), IEEE International Conference, Barcelona, 2011, p. 1623.1630.

[56] F. Brunet, A. Bartoli, and R. Hartley, “Monocular template.based 3d surface reconstruction: Convex inextensible and nonconvex isometric methods,” CVIU, p. 157.186, 2014.

[57] D. Pizarro, A. Bartoli, and T. Collins, “Isowarp and conwarp: Warps that exactly comply with weak-perspective projection of deforming objects.,” BMVC, 2013.

[58] R. Khan, D. Pizarrou, and A. Bartoli, “Schwarps: Locally projective image warps based on 2d schwarzian derivatives,” ECCV, 2014.

[59] D. Pizarro, R. Khan, and A. Bartoli, “Locally projective image warps based on 2d schwarzian derivatives,” International Journal of Computer Vision, p. 93.109, 2016.

[60] D. Ngo, J. Ostlund, and P. Fua, “Template-based monocular 3d shape recovery using laplacian meshes,”

[61] T. Ngo, D. Östlund, J. O., and P. Fua, “Template-based monocular 3d shape recovery using laplacian meshes,” Pattern Analysis and Machine Intelligence, p. 172.187, 2016.

[62] K. Weinberger and L. Saul, “Unsupervised learning of image manifolds by semidefinite programming,” Int. J. Comput. Vision, 70(1):77.90, 2006.

[63] S. Hosseini and A. H, “3d estimation of isometric surfaces using a tof-based approach,” ISVC, 2014.

[64] J. Fayad, A. DelBue, and L. Agapito, “Non-rigid structure form motion using quadratic models,” BMVC, 2004.

[65] R. Hartley and A. Zissman, Multiple View Geometry in Computer Vision. 2004.