Deformable and Articulated 3D Reconstruction from monocular video sequences

Marco Paladini

Queen Mary, University of London

Submitted for the degree of Doctor of Philosophy

Queen Mary, University of London

2012

Deformable and Articulated 3D Reconstruction from monocular video sequences

Marco Paladini

Queen Mary, University of London

Abstract

This thesis addresses the problem of deformable and articulated *structure from motion* from monocular uncalibrated video sequences. Structure from motion is defined as the problem of recovering information about the 3D structure of scenes imaged by a camera in a video sequence. Our study aims at the challenging problem of non-rigid shapes (e.g. a beating heart or a smiling face). Non-rigid structures appear constantly in our everyday life, think of a bicep curling, a torso twisting or a smiling face. Our research seeks a general method to perform 3D shape recovery purely from data, without having to rely on a pre-computed model or training data. Open problems in the field are the difficulty of the non-linear estimation, the lack of a real-time system, large amounts of missing data in real-world video sequences, measurement noise and strong deformations. Solving these problems would take us far beyond the current state of the art in non-rigid structure from motion. This dissertation presents our contributions in the field of non-rigid structure from motion, detailing a novel algorithm that enforces the exact metric structure of the problem at each step of the minimisation by projecting the motion matrices onto the correct deformable or articulated metric *motion manifolds* respectively. An important advantage of this new algorithm is its ability to handle missing data which becomes crucial when dealing with real video sequences. We present a generic bilinear estimation framework, which improves convergence and makes use of the manifold constraints. Finally, we demonstrate a sequential, frame-by-frame estimation algorithm, which provides a 3D model and camera parameters for each video frame, while simultaneously building a model of object deformations.

Submitted for the degree of Doctor of Philosophy

Queen Mary, University of London

2012

Contents

Acknowledgements

This research has received funding from the European Research Council under ERC Starting Grant agreement 204871-HUMANIS.

I would like to thank everyone who contributed to my PhD years, to the making of the thesis, to all the good times. Thanks to my adviser, Dr Lourdes Agapito, for the hard work in teaching me everything I know about computer vision, and for much more than just that. She taught me how to be a scientist, what research is about and how to present work at conferences. She taught me how so many seemingly different research have so much in common. I have been very lucky to start a PhD with her. I thank Lourdes also for the great help and support during times of struggle, for her listening, for the dedication and effort she puts towards the well-being of everyone in her group.

I'd like to thank my family, my Mum and Dad who have always supported me in everything, and my sister Stefania and her husband Salvatore. Thanks for always being there for me.

Special thanks to my fiancee Maria Dolores, who has seen all my struggling with the PhD, the thesis, the research, the life in London... and she still loves me. Thank you my love.

I have been very lucky to have had collaborations with some brilliant researchers, and co-authoring papers with them. Thanks to Alessio Del Bue, for sharing his in-depth experience on experiments in non-rigid reconstruction, and for his organisation skills and focus. I thank Adrien Bartoli for his teachings and great insight. Thanks to João Xavier for his genuine interest in the variety of math problems we had.

Thanks to my flatmate: Francesco Russo. Always a good friend, he had his own PhD life, so I was never alone in all the difficulties. I thank him also for pushing me to come to London in the search for a PhD. And I'm grateful to Andrew Davison, who pointed me to apply for a PhD with

Lourdes here at Queen Mary.

I'd like to thank everyone in Queen Mary. I've had a great environment, full of bright people, fun, teaching, seminars, social events, training... it's a great experience. Thanks to all the staff (Sue, Julie, Melissa...) for being always helpful with any administration issues. Many thanks to the system support staff (sometimes called "system guys", or just "systems" for short) for showing me some real sysadmin work done. I could learn a hell of a lot from the likes of Lukasz, David, Tim, Keith, Matt, and all the others. Thanks for the nightly back-ups, the virtual machines setup, the server upgrades... I know I'll hardly find better run systems anywhere.

Thanks to the lecturers, it was great to do some teaching in the labs. Thanks to Fabrizio for some good Italian coffee, thanks to Paul Curzon for passing on to us his passion about teaching, he is right when he says "anyone can learn computer programming". I also want to thank all the fellow students and research assistants here for being good friends and always ready to help one another. Also, lunch together is probably the most creative time of the day. Thanks to all of you (non-exhaustive list): Cavan, Brian, Matteo, Khalid, Milan, Samuel, Stuart, Colombine, João, Ravi, Nikos, Tassos, Francesco, Chris... and all of those I meet down the pub. Thanks to my friends from Naples, Italy: (Dino and Loredana, Daniela, Alessia, Francesco "il coniglio", Claudia, Vincenzo "il foggiano", Mimmo, Ilaria, Carlos, Giovanni...) and to all the ones who studied physics with me in Naples now scattered around the world.

Related publications

- Marco Paladini, Alessio Del Bue, Marko Stošić, Marija Dodig, João Xavier, Lourdes Agapito **“Factorization for non-rigid and articulated structure using metric projections”**, in the IEEE Conference on Computer Vision and Pattern Recognition, June 2009, Miami, Florida.

- Marco Paladini, Adrien Bartoli, Lourdes Agapito **“Sequential Non-Rigid Structure-from-Motion with the 3D-Implicit Low-Rank Shape Model”**, at the 11th European Conference on Computer Vision, September 2010, Crete, Greece.

- Alessio Del Bue, Joao Xavier, Lourdes Agapito, Marco Paladini **“Bilinear Factorization via Augmented Lagrange Multipliers”** , at the 11th European Conference on Computer Vision, September 2010, Crete, Greece.

- Marco Paladini, Alessio Del Bue, João Xavier, Lourdes Agapito, Marko Stošić, Marija Dodig **“Optimal Metric Projections for Deformable and Articulated Structure-From-Motion”**. International Journal of Computer Vision, pp. 1-25, 2011.

- Carme Julià, Marco Paladini, Ravi Garg, Domenec Puig, Lourdes Agapito **Automatic estimation of the number of deformation modes in non-rigid SfM with missing data** , at SCIA 2011 - Scandinavian Conference on Image Analysis, Ystad Saltsjöbad, Sweden 23-27 May 2011.

Chapter 1

Introduction

1.1 Introduction

The recovery of 3D scene information from video sequences has long been at the core of computer vision. In recent years a great variety of algorithms and techniques have been proposed for the reconstruction of 3D shape from uncalibrated video sequences. It is possible in principle to perform such reconstructions from an image pair taken by two cameras from different viewpoints, or by a single moving camera. In the case of a single camera (*monocular* video sequences), if the motion of the camera were known (i.e. if it is attached to a precisely-driven robot arm) then calculating depth would be a simple matter of triangulation. In the more general uncalibrated case, the camera motion itself is also uncertain. The problem of combined inference of the 3D motion of a camera and the geometry of the scene it views is generally known as *Structure from Motion* (SFM).

The fundamental assumption which has allowed solutions to the structure from motion problem to be achieved is that of scene rigidity. Our research is aimed at the more challenging problem of non-rigid reconstruction from a video sequence taken by a single camera. This problem is known as *Non-Rigid Structure from Motion* (NRSfM). The goal of NRSfM is to infer the 3D

shape of a deformable or articulated object when the camera position and its internal parameters are all unknown.

Progress in this field has been primarily motivated by its wide-ranging applications in areas such as human-robot and human-computer interaction, surveillance, athletic performance analysis, medical imaging, computer animation for the games and film industries and augmented reality.

In the case of human motion, motion capture systems exist which can recover body movements using multiple synchronised cameras. However these systems often rely on markers, for example reflective surfaces that have to be attached to the body. Other alternatives include the use of motion sensors, which are costly and technically complex, furthermore the person must wear them, which results in unnatural movements.

Articulated motion recovery has also been formulated as a structure from motion problem. The goal is to perform 3D reconstruction of the segments of an articulated body, together with the position of rotation axis and joint angles. The capture of articulated motion using markers is not viable for commercial applications such as video-games, or human robot interaction. The animation industry is moving away from markers based solutions and embracing marker-less approaches.

Our research focuses specifically on recovering 3D shape of deformable and articulated objects from video sequences acquired with a single camera. Moreover, we seek methods able to recover the shape of a generic object, when no pre-defined 3D model is available. We adopt a data-driven approach, in which both 3D shape and deformation models are obtained purely from data.

1.2 The rigid case

Structure from motion (SFM) can be defined as the problem of estimating both the motion of a camera and the 3D geometry of the scene it views solely from a sequence of images. Commonly, SFM methods aim at inferring 3D structure purely from 2D correspondences of feature points established throughout a sequence. When two (or more) calibrated views of the same object are available, 3D information can be recovered via triangulation [50]. In most interesting scenarios,

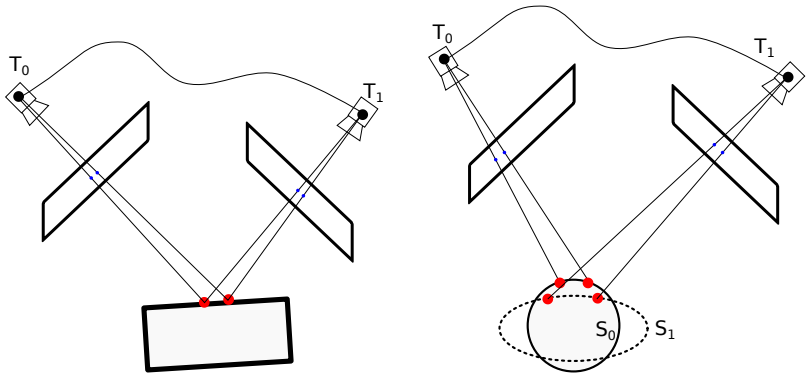
however, calibration is not available: the camera motion and its internal parameters have to be estimated from feature correspondences. It was shown by Longuet-Higgins [65] that such reconstruction is possible for a single camera taking images of a rigid object from different locations. Self-calibration methods have been devised which allow camera parameters to change during the video sequence following on from the seminal work of Faugeras *et al*. [40].

The factorisation method proposed by Tomasi and Kanade [110] has been one of the most influential works in structure from motion. It recovers 3D shape form a monocular video sequence assuming an orthographic camera projection model. The orthographic camera model is an approximation of the more general perspective camera model, suitable when the relief of the object is small compared to its distance from the camera. The use of an affine camera model allows the factorisation algorithm to reconstruct the 3D structure and camera motion consistent with feature tracking data in all the frames with a linear method.

The factorisation method was extended to the case of multiple independent moving objects by Costeira and Kanade [28]. A factorisation approach is possible also in the case of the projective camera model as shown by Sturm and Triggs [104]. Perspective reconstruction was achieved by defining and computing an additional unknown for each point, called the *perspective depth*.

The reconstruction of rigid scenes is now a well understood problem, with a wide variety of real-world applications in many different areas from robot navigation to cinema post-production. The success of the factorisation algorithm for rigid SFM due to its simplicity sparked interest in the community to extend it to the case of non-rigid motion.

Non-rigid structure from motion seeks to relax the rigidity assumption and reconstruct the time-varying 3D shape of an object. In this thesis we focus on the challenging problem of recovering deformable and articulated objects from video sequences taken by a single camera. Both the scene structure and camera movements are not known beforehand. We seek to model generic objects, with a goal of recovering both a model for the deformations and the non-rigid 3D shape purely from 2D correspondences.



(a) Rigid case (b) Deformable case

Figure 1.1: Triangulation from two frames of a video sequence: The camera moves around the object. If the object is rigid the triangulation problem is well posed, but it is under constrained if the object is deforming.

1.3 Non Rigid Structure from Motion

In the case where the shape of the object in the scene changes over time, reconstructing the 3D position of a feature point from two different images is an ill-posed problem, as shown in Figure 1.1. An object is deformable if relative point positions are not constant during the video sequence. Take for example a smiling face or a beating heart, feature points tracked on the surface of such objects do not move rigidly in space. This simple consideration makes it clear that the problem is equivalent to reconstructing from a single image. The non-rigid structure from motion problem, is thus an ill-posed problem by its very nature.

The key insight that has allowed the reconstruction of deformable scenes is the assumption that deformations are not arbitrary: points in 3D move together under the effect of physical forces. For most real world objects, the deformations can be modelled as small displacements from a mean shape. Figure 1.2 shows the popular low-rank basis shapes model introduced by Bregler *et al*. [15], where the shape configuration is explained as a linear combination of a set of modes of deformation, or bases, each weighted by time-varying coefficients. This model has proven successful in reconstructing many real-world objects, in a factorisation framework, where both the model and the coefficients are unknown. This linear model allows a factorisation approach

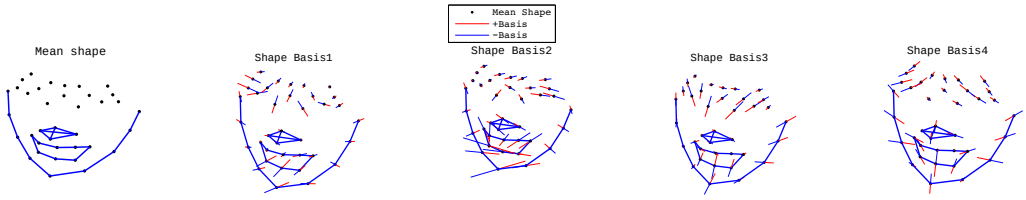


Figure 1.2: Example of low-rank basis shapes model: the average rigid shape can deform according to the linear combination of a fixed set of deformation modes. The figure shows directions where the deformable points can move, with a positive or negative coefficient applied to the basis.

for the non-rigid structure from motion problem.

Non-rigid factorisation uses the assumption of a low-rank basis shapes model to express the feature point tracks as the product of a motion matrix, expressing time-varying camera pose and model coefficients, with a shape matrix, encoding the possible modes of deformation. Unfortunately, this approach results in a non-linear estimation problem. The non-linear constraints on camera pose, together with the non-linearities induced by the mixing of shape coefficients and camera parameters make the estimation problem difficult. Most of the research in this field is aimed at solving the estimation problem. Bregler *et al*. [15] formulate the reconstruction in a linear way, by computing an affine decomposition using singular value decomposition (SVD) and then computing an invertible upgrade matrix to enforce the constraints on the camera matrices (called *metric constraints*). It was shown by Brand [12] that such linear methods are prone to fail in the presence of noise. Xiao *et al*. [128] show theoretically that linear methods would fail to cope with noise, due to the process of computing an upgrade matrix. They show that more constraints are needed to be able to solve the problem in closed-form, and propose a linear method to exploit such constraints. Also, a closed-form solution based on SVD does not provide a result for the case of missing data: when 2D feature points go out of view in the image, or when features are occluded or not tracked successfully. Various approaches that do not compute an upgrade matrix try to solve the non-linear estimation problem directly [112, 1, 35, 36, 8, 111]. However, they also make use of additional constraints in order obtain robust solutions. Imposing

Figure 1.3: Motion capture with reflective markers require expensive infrared cameras and complicated setup

for example smoothness priors [112, 1, 36], or statistical priors [111, 8], or priors on the rigid component [35]. Imposing such additional constraints allows solutions to be robust to noise and missing data.

Our first contribution deals with the difficulty of imposing the metric constraints. We propose an algorithm to enforce the metric constraints without computing an invertible upgrade matrix. We demonstrate its robustness to noise and missing data in the measurements, even without the imposition of additional smoothness or statistical priors. We contribute a speed-up computation method that makes use of smoothness priors, when such prior is applicable. We extended this idea into a general framework for bilinear estimation with manifold constraints. In addition, we contribute a novel formulation suitable for sequential frame-by-frame non-rigid structure from motion, which breaks free from the common requirement of all current methods of processing the data in batch.

1.4 Motivation

The pursue of this research is motivated by recent progress in the areas of deformable and articulated motion recovery and by the need for model free and marker free approaches. The human body shows great variety of deformations and articulated motion, research on human motion recovery is of great interest and with a very active research community.

With no doubt success in solving open problems in this field would lead to many useful applications, marker-less motion tracking for computer graphics, video analysis for various applications: from medical to surveillance. We also see a possible application of this project in the field of humanoid robotics, where the recovery of the 3D human motion can be used to train a humanoid robot, controlled by the movements of a human operator. Finally, we see applications in the field of human-robot and robot-robot interaction, where the motion data can be used to coordinate the work of the interacting agents.

We propose novel algorithms to advance the field of non-rigid structure from motion to overcome the current challenges such as the ambiguities in the non-linear estimation, the lack of a real-time system, and the ability to deal with large amounts of missing data in real-world video sequences. Those efforts are also directed at eliminating the infrared markers currently used in motion capture systems available today. Those systems are not only very expensive and difficult to use, they also require a complicated setup, as can be seen in Figure 1.3. The person has to wear special clothes and put reflective markers on it that will be tracked using infrared cameras. In the example in the Figure, the system is composed of 12 cameras. Our research aims at finding a solution to the more challenging problem of monocular reconstruction. In many cases only one camera is available, for example, in post-processing movies and television recordings, in laptops and digital phones, consumer cameras, and in medical imaging such as laparoscopy, where only one camera is available to capture images.

Figure 1.4: Motion capture systems applied in the movie industry: infrared markers are used together with a multi-camera system of infrared cameras to track the position of the body, while a camera focused on the face captures facial expressions, to be re-targeted on the 3D animation character. Images copyright Twentieth Century Fox.

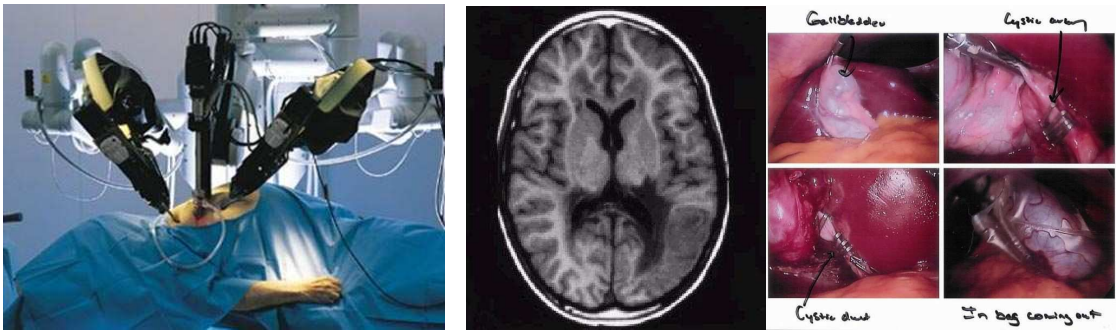


Figure 1.5: Medical imaging applications of non-rigid structure from motion vary from robotic surgery (Left), brain MRI scan (Middle), to laparoscopy (Right). Images copyright (left to right) Intuitive Surgical, Massachusetts General Hospital Center for Morphometric Analysis and Paodavy Medical Services.

1.5 Applications

In recent years, the movie industry has shown great interest in the techniques and methods that computer vision provides for reconstructing the shape and motion of deformable and articulated objects. One of the motivations for motion capture in the movie industry lies in the application called augmented reality. The technique consists in capturing camera motion in order to insert a virtual object in the scene, such as rendering a computer graphics model onto the image, as shown in Figure 1.6. The knowledge of the camera position is crucial for the virtual object to perform a realistic trajectory in the final movie. Another rationale for capturing motion per-

Figure 1.6: Another example of motion capture system applied to movies. This augmented reality application consist in capturing the body movement of the actors with the help of infrared markers. Such movements are augmented with a series of 3D animation models. All occlusions must be handled manually by the graphic artists. Images copyright Walt Disney Pictures.

formed by actors is motion re-targeting, as shown in Figure 1.4. The work by graphic artists to animate the virtual character is greatly reduced, if the facial expressions of the actors are captured.

Motion re-targeting is not only useful for movies, but also for robotics applications. Figure 1.7 shows an example application of motion re-targeting. Joint angles describing a body posture are captured from gyroscopes attached to the body, the capture angles can be replicated by the motors of a humanoid robot. This is particularly useful in machine learning scenarios, where an operation could be performed by a human multiple times, allowing a model for the operation to be built, and given to the robot motors for execution.

Figure 1.5 shows possible medical applications of this research. A robot surgeon could provide detailed 3D models of moving tissues or organs using video cameras attached to the robotic arms. The analysis of brain with magnetic resonance imaging (MRI) can be automated by building a model of the variations that exist among individuals, which can be thought as a non-rigid estimation problem. A medical scenario where only one camera is available to analyse deformable tissues is laparoscopy, NRSfM can provide the 3D shape and size of the organs where the surgeon is operating. Figure 1.8 shows an example of augmented reality without any

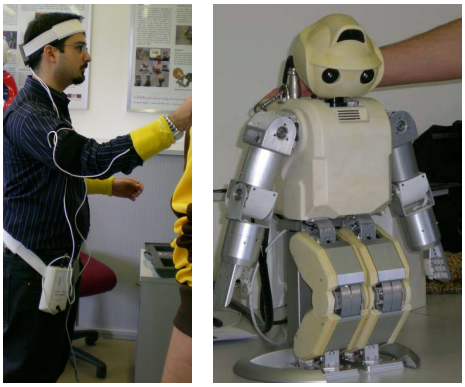


Figure 1.7: Left: Example of motion capture sensors, Right: Humanoid robot replicating captured motion

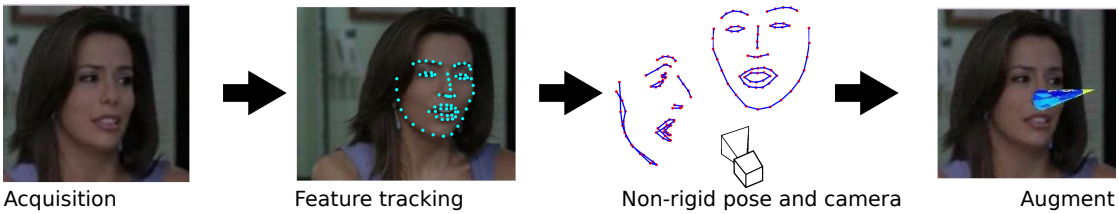


Figure 1.8: Pipeline of an augmented reality application. Following image acquisition, feature point tracking detects correspondences between image points. Non-rigid structure from motion can estimate the 3D shape and camera position for every frame. The final step is the insertion of a virtual object, which will follow the movement that has been captured.

markers, using a non-rigid structure from motion approach. The video is analysed to detect feature correspondences between frames, those features are fed to a non-rigid structure from motion method to estimate 3D shape and camera position at every frame. Capturing the camera movement allows the insertion of virtual objects in the scene. Capturing deformations also allows computer generated 3D graphics to follow a realistic motion.

1.6 Contributions

In our work we explore a new unified approach to deformable and articulated structure from motion. None of the methods proposed so far has focused on the computation of motion matrices that satisfy the metric constraints exactly, but only in a least squares sense. Therefore, the

recovered matrices are not guaranteed to satisfy the constraints when data is affected by noise or missing tracks. Most non-linear methods enforce metric constraints through parametrisation. Although this ensures the metric constraints are satisfied, additional priors are usually required in order to avoid local minima, and to improve robustness to noise. We show that dealing with metric constraints through projection can provide state-of-the art results without using additional priors.

- We contribute an algorithm that provides the global optimum in projecting a candidate motion matrix into the manifold of metric solutions. With this approach, we enforce the non-linear constraints on the motion matrices. Similarly, in the case of articulated shapes, we efficiently compute the joints, given the non-linear constraints on the motion of the two bodies. The result is an algorithm where the recovered motion matrices have the exact orthogonality constraints imposed. One of the main advantages of this approach is the ability to handle a large amount of missing data, as we demonstrate experimentally.

- We proposed a novel optimisation method based on augmented Lagrange multipliers where the manifold constraints are decoupled from the bilinear estimation problem, which is common in articulated, rigid, and non-rigid structure from motion. In addition, the proposed optimisation scheme obtains better speed and convergence compared to other state of the art methods, and is not limited to those problems. In fact, bilinear estimation with manifold constraints is a problem that appears frequently in computer vision and in other fields.

- We propose a novel implicit model and an algorithm designed to use video images as they become available, in a sequential estimation framework. A 3D model of the non-rigid object in the scene is obtained for each image frame, while simultaneously building and updating a model for the deformations. This new technique moves the non-rigid structure from motion problem in the direction of real-time estimation of 3D shape, camera parameters, and modelling of non-rigid objects, from a monocular video sequence. To the best

of our knowledge, we present the first method that can model 3D objects frame by frame, without having to analyse the entire sequence, and without relying on any a-priori model of the scene.

The contributions of this thesis are presented as follows. Chapter 2 will discuss the literature on 3D shape recovery from monocular video, discussing the wide variety of methods that have been proposed for deformable and articulated reconstruction, focusing on factorisation approaches to structure from motion methods. We provide a taxonomy of methods for non-rigid shape reconstruction where we divide approaches according to the shape model used and to the optimisation technique employed to estimate the parameters. Chapter 3 details our Metric Projections algorithm, an alternating approach to solve for non-rigid 3D shape and motion, associated with a globally optimal projection step of the motion matrices onto the manifold of metric constraints. Chapter 4 describes a generalised framework for solving a large class of bilinear problems in computer vision with manifold constraints. Chapter 5 describes our new sequential approach to non-rigid structure from motion in which the 3D model is built sequentially in a frame-to-frame fashion. Finally, Chapter 6 presents the closing discussion of this dissertation, introducing possibilities for further work to advance the field of non-rigid structure from motion.

Chapter 2

Literature Review

The recovery of 3D structure information from image sequences is a fundamental problem in computer vision. The goal is to estimate the 3D coordinates of scene points captured on video. This problem has been largely studied and many viable solutions have been found in the case where the scene is rigid. Our research focuses on the more difficult problem of 3D recovery when the object in the video is non-rigid, that is, its shape can change through time by deforming or articulating. We seek 3D models to express the time-varying shape of the objects in the image.

The video is acquired by a camera, which can be seen as a projective device: each point in space is projected onto a point on the image plane. Often the position and orientation of the camera and its internal parameters are also unknown, and thus need to be estimated. The goal of uncalibrated 3D structure recovery is formulated as the joint estimation of the 3D position of the points in space and the pose and internal parameters of the camera. This problem is known as ”Structure-from-Motion”. This chapter discusses the literature in the field of 3D reconstruction from image sequences focusing on the case of non-rigid shape recovery. We will pay special attention to the class of methods central to our research, namely factorisation approaches, starting with the well established results on rigid shapes and progressing to current research in deformable and

articulated structure.

2.1 Approaches to 3D shape reconstruction

A vast amount of different techniques have been proposed in computer vision to deal with the problem of reconstructing 3D shapes from video under different conditions such as different number of cameras or types of scenes, known or unknown calibration, etc. Different visual cues have been used in the literature to infer the shape information present in images. Such inference can be based on shading, silhouettes, texture, focus, motion, or other visual cues. In this dissertation we are interested in *Structure from Motion* approaches which use the motion present in the image as the only cue to estimate the 3D scene geometry and the motion of the camera.

This chapter is organised as follows. First we give an overview of techniques and methods that use cues other than motion to recover shape information from a single image or a monocular video sequence. We then discuss research in *Structure from Motion* focusing on the factorisation algorithm for rigid scenes, given its significance to non-rigid structure from motion as the approach that allowed its first formulation. We will then review the literature in non-rigid structure from motion providing a taxonomy of the approaches proposed so far, classifying them according to the deformation models and to the optimisation techniques they use. We focus on methods that have followed the prevalent factorisation formulation using the low rank deformable shape model and emergent techniques that tackle the problem using alternative shape models, optimisation techniques or different priors.

2.1.1 Shape-from-X

Many cues in the image are directly related to the 3D shape of the objects in the scene. The wide array of methods for performing shape recovery is generically known as “Shape-from-X”, where “X” can in turn be “Shading”, “Texture”, “Silhouettes”, “Focus” or others. These methods are fundamentally different from the *Structure from Motion* approaches we will study in detail

because they use cues other than motion to infer the 3D structure of the scene.

Shape from Shading uses the light source location, and surface reflectance (usually assumed to be Lambertian) to recover surface normals. This method is biologically inspired: shading conveys depth information to the observer of a painting. Lambertian surfaces reflect light in all directions, therefore the brightness of a surface point can be expressed as the scalar product of the light direction vector and the surface normal vector at that point, multiplied by the surface albedo, and a constant representing the intensity of the light source. Initially formulated by Horn [57] in 1970, it is now a mature field with a rich literature [135].

In *Photometric Stereo* [126] the surface normals and reflectance properties of an object can be recovered using multiple images taken from the same viewpoint but acquired under variable lighting conditions. When a Lambertian surface model and single point-like light sources are assumed in each image, three or more images taken with different lighting direction provide enough constraints to recover the surface normals and the light direction vector. Basri *et al*. [10] recently proposed a solution to the case of general, unknown and unconstrained lighting, relaxing the assumption that a single point-like light source should be present in each image. This work is based on the result that general lighting conditions can be represented using low order spherical harmonics and allows to frame photometric-stereo as a factorisation problem with constraints on one of the factors.

Photometric Stereo techniques normally assume a rigid object in the scene. Hernández *et al*. [53] proposed a method for non-rigid reconstruction based on coloured lights. The acquisition setup consists of three coloured light sources (red, green and blue) with different lighting directions. The three colour channels of each image provide enough constraints to reconstruct the time-varying 3D shape. Surface normals are recovered for each frame and combined with 2D optical flow to register them over time to generate a single deforming 3D surface.

Shape from Silhouettes, introduced by Laurentini [62], estimates the shape of an object from multiple images of its silhouette taken from different viewpoints. Assuming each view is taken with a calibrated camera, each 2D silhouette can be back-projected to give a generalised cone

or a volume in which the 3D object must lie. The intersection of the back-projected silhouettes taken from different viewpoints provides a 3D reconstruction of the object known as the visual hull, which is in fact a bound for the true geometry of the 3D object. Usually formulated for rigid objects, Cheung *et al*. [22] were the first to extend this idea to articulated shapes, in particular to human body pose.

Considered a generalisation of shape from silhouettes techniques, the *Space Carving* algorithm, introduced by Kutulakos and Seitz [61], can perform the reconstruction of an arbitrarily shaped 3D scene viewed by a set of calibrated cameras placed at arbitrary positions when no information is available about any specific features or their correspondence. The volume is represented as a set of voxels in 3D space and the algorithm iteratively *carves* out the shape of the scene by removing voxels that are not photo-consistent with the images at each iteration. A voxel is photo-consistent when the colour predicted by the radiance function is the same in all the images in which it is visible. The *Space Carving* algorithm reconstructs the *photo-hull* of a set of images, also defined as the *least commitment reconstruction*, that is, a 3D reconstruction photo-consistent with the images that does not make any assumptions about the geometry of the scene.

2.2 Rigid Structure from Motion

Structure from motion (SfM) or multi-view reconstruction can be defined as the problem of combined inference of the motion of a camera and the 3D geometry of the scene from a sequence of uncalibrated images using as input only the 2D image coordinates of a number of features which can be matched through the sequence. The fundamental assumption which has allowed robust solutions to be achieved is that of scene rigidity: if objects are known not to change or deform, their shapes are invariant entities of which estimates can be gradually refined. Large numbers of well-localised features of high image salience — usually “corner” points or lines — are detected in each image of a video sequence. The features that are associated with the same 3D point in space are then matched between each pair of consecutive (or close) video frames. The assumption of rigidity in the scene is then used to assert that the change in image

position of features from one frame to the next is due purely to the movement of the camera relative to the unknown but static 3D geometry or "structure" of the features. This translates into mathematical constraints on the parameters describing camera motion, and many feature matches provide enough constraint equations for solutions for both the motion and the locations of the 3D features to be obtained.

The estimation of these 2D correspondences in an image sequence remains an open problem in computer vision, with a wide range of approaches [134, 101, 67]. In this thesis we will not focus on solving the matching problem and instead we will make the assumption that matching data is available to perform 3D reconstruction.

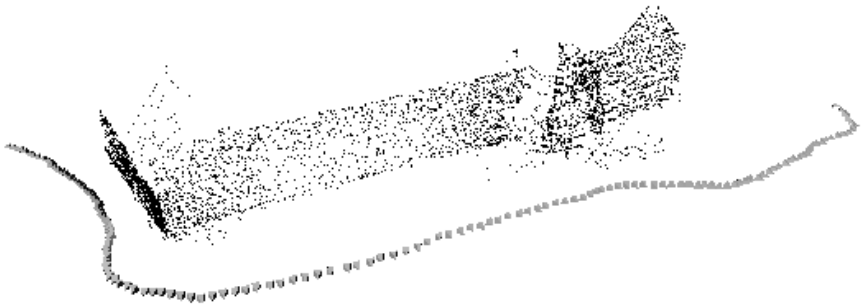


Figure 2.1: Structure from motion (SfM) pipeline: from 2D matching to 3D modelling. Results of 3D reconstruction of a large building and camera pose estimation from a sequence of images with varying camera intrinsics. From Pollefeys and Van Gool [92].

2.2.1 Uncalibrated case

The recovery of 3D information from 2D feature correspondences in an image sequence can be performed via triangulation when two (or more) calibrated views of the same scene are available [50]. A more interesting and practical scenario is when the camera used is uncalibrated: its internal parameters such as the focal length, etc. are not known in advance. These techniques which work even when the camera used is uncalibrated are known as self-calibration algorithms in the literature and have provided the flexibility of being applicable even in cases where little is known about the details of image capture. In early work, Ullman [118] first proved that

simultaneous camera calibration and 3D reconstruction is possible in the case of rigid scenes. This was then followed by the seminal work of Faugeras *et al*. [73] which established the theory of self-calibration and provided practical algorithms. The simple assumption that the camera used has fixed focal length over time provided enough information for self-calibration, provided that the camera motion is "general" — it exercises all of its degrees of freedom [51, 115].

Research in structure from motion in the $90's$ was then dominated by providing solutions to this problem of joint calibration and 3D reconstruction adjusting it to specific scenarios that needed special solutions such as when the camera is known only to rotate about its optical centre [52, 3]; only to translate without rotation [75]; or allowing it to deal with the most flexible case of a camera equipped with a zoom lens so its focal length could vary [91, 56].

In [92], Pollefeys and Van Gool provided one of the first complete structure from motion (SfM) pipelines: from 2D matching to 3D reconstruction of a mesh model of the scene from long sequences acquired with an hand-held uncalibrated zooming camera with varying intrinsics. The reconstruction was built incrementally: a pair of images was first chosen and a projective 3D reconstruction obtained. For each new image, the camera pose was estimated relative to this reconstruction and the reconstruction updated with the new data. The final reconstruction was upgraded to metric using their self-calibration algorithm that can deal with varying intrinsics [91] followed by a final non-linear refinement of all the parameters. This pipeline was successfully applied to recovering 3D models of ruins in archaeological sites or to large buildings as shown in Figure 2.1.

2.2.2 Bundle adjustment

Most 3D reconstruction methods ultimately rely on a final large non-linear optimisation to provide a joint refinement of the 3D coordinates of all the observed points, as well as the camera parameters (pose and calibration) for all the frames. This is achieved by minimising the squared image reprojection error between the image locations of observed and predicted image points in all the views in which they are visible. Optimising image reprojection error gives a maxi-

mum likelihood estimate of the parameters, provided the noise in the image measurements is Gaussian. This joint optimisation of 3D structure and camera parameters is known as *bundle adjustment* in the literature [116] and was initially conceived in the field of photogrammetry during the 50s. Naturally, bundle adjustment requires a good initial estimate of the 3D structure, camera pose and calibration parameters for it to converge to the global minimum and not be trapped in a local one. Much of the research has therefore focused on providing closed form solutions both in the calibrated [81] and the uncalibrated case [91] that provide good initial estimates to the non-linear optimisation.

Typical structure from motion problems might involve thousands of 3D points in hundreds of frames which amounts to a very large number of parameters to be estimated. The Gauss-Newton optimisation generally used for this non-linear least-squares problem requires the inversion of a Hessian matrix that has the same dimensions as the number of unknowns. Since this number can be huge, bundle adjustment algorithms make use of the sparse nature of this matrix to make the problem tractable. Each error term associated with a 2D observation only depends on a very small number of variables: the 3D coordinates of the point and the camera parameters of the frames in which it is visible. Fortunately, the inversion of the Hessian can be hugely speeded up by taking advantage of its block diagonal nature. A public implementation of bundle adjustment has been developed by Lourakis and Argyros [66].

Much of the recent progress in structure from motion has come from improving bundle adjustment's efficiency, to improve its performance and make it amenable to the real-time domain, and its scalability, to deal with very large Internet-based data sets with hundreds of thousands of images.

Reconstruction of large-scale data sets acquired from community photo-collections was demonstrated by Snavely *et al*. [102] for the purpose of image-based rendering, to provide the user with a virtual tour of a scene. Figure 2.2 shows more recent results by Agarwal *et al*. [4] of 3D reconstruction of famous buildings, such as the Colosseum in Rome, or even whole cities, such as Dubrovnik, from large sets of uncalibrated photographs downloaded from Flickr. The

system, which uses distributed matching and reconstruction algorithms and is designed to maximise parallelism in every stage of the pipeline, is able to process 150000 images in 24 hours on a cluster with 500 cores. On the other hand, real-time methods have now allowed to map a small workspace with one handheld camera [60], to quickly construct 3D models of small objects using a web-cam [87] or even to obtain live dense 3D models using current desktop hardware with GPUs [79, 80].

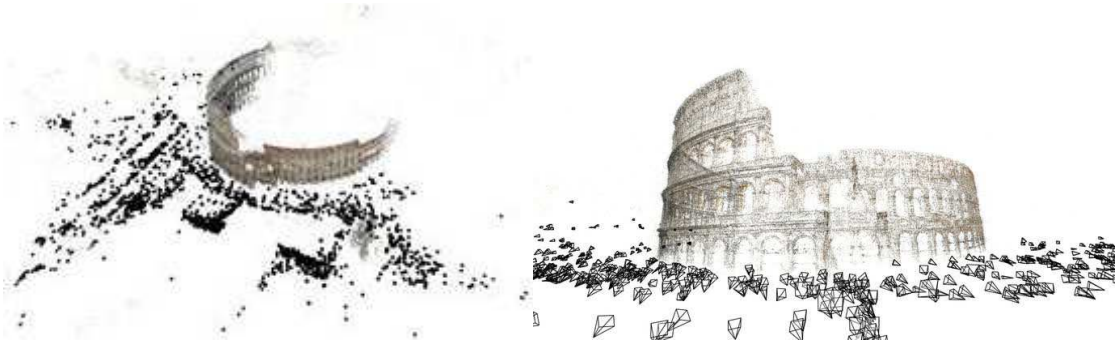


Figure 2.2: Large-scale reconstruction. The famous Colosseum building in Rome is reconstructed from Fickr community photo-collections, together with other famous buildings of Rome. Reconstruction results from the "Building Rome in a Day" project by Agarwal *et al*. [4]

2.2.3 Factorisation

In the common scenario when the affine camera model is a good approximation of the image capture process — when the relief of scene objects is much smaller than their distance from the camera — a linear algorithm that provides the Maximum Likelihood Estimate (MLE) of both 3D structure and camera motion over long sequences can be used. Due to its elegance and simplicity, Tomasi and Kanade's *factorisation algorithm* [109] has been one of the most influential works in structure from motion.

The factorisation algorithm is a batch method: it stacks the 2D coordinates of all the matched features in all the frames into a large *measurement matrix*, processing all the frames simultaneously instead of incrementally as other SfM pipelines [92]. The key insight is that this matrix is rank deficient (in in the case of rigid scenes the rank is at most 3) and so singular value de-

composition can be performed to recover the shape and motion components. The factorisation algorithm is simple but powerful: it is optimal, linear (therefore fast) and processes all the frames simultaneously. Due to all these advantages it is often used as the initialisation to a final bundle adjustment optimisation.

Amongst its disadvantages, the original algorithm assumes affine cameras and requires that all the points are viewed in all the frames. However, numerous extensions have been proposed for the cases of para-perspective and then perspective cameras, multiple independently moving objects; the use of various image features other than corners such as lines and line segments and to the case of incomplete observations. Crucially to the work presented in this thesis, the factorisation framework has also been extended to deal with non-rigid scenes in the case of articulated and deformable motions. Therefore, in the next sections we describe Tomasi and Kanade's *rigid factorisation algorithm* [109] in detail before we go on to describe its formulation for non-rigid structure recovery.

Tomasi and Kanade's factorisation algorithm

Consider the set of 2D image trajectories obtained when the points lying on the surface of a 3D object are viewed by a moving camera. Defining the non-homogeneous coordinates of a point j in frame i as the vector $\mathbf{w}_{ij} = (u_{ij}\ v_{ij})^T$ we may write the measurement matrix $\mathtt{W}$ that gathers the coordinates of all the points in all the views as:

$$\mathtt{W} = \begin{bmatrix} \mathbf{w}_{11} & \dots & \mathbf{w}_{1P} \\ \vdots & \ddots & \vdots \\ \mathbf{w}_{F1} & \dots & \mathbf{w}_{FP} \end{bmatrix} = \begin{bmatrix} \mathtt{W}_1 \\ \vdots \\ \mathtt{W}_F \end{bmatrix} \tag{2.1}$$

where F is the number of frames and P the number of points. This matrix of size $2F \times P$ contains all the projections of feature points. It is possible to decompose $\mathtt{W}$ into the product of two matrices:

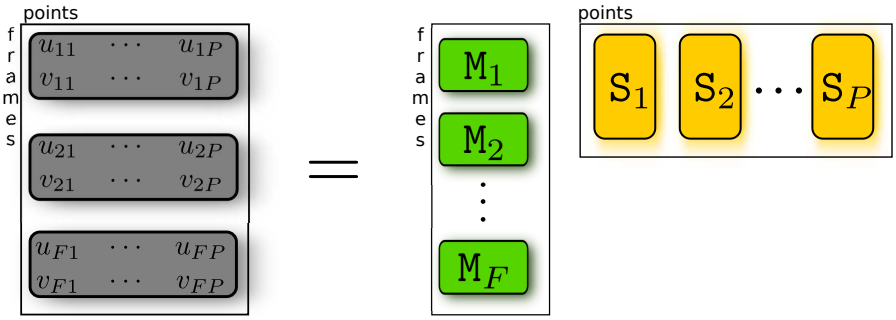


Figure 2.3: The measurement matrix containing the projection of all points in all frames is decomposed into a series of motion and shape components.

$$\mathtt{W} = \mathtt{MS} \tag{2.2}$$

Where $\mathtt{M}$ and $\mathtt{S}$ correspond to the motion and shape components of the measurement matrix.

Matrices $\mathtt{M}$ and $\mathtt{S}$ can be expressed as:

$$\mathtt{M} = \begin{bmatrix} \mathtt{M}_1 \\ \mathtt{M}_2 \\ \vdots \\ \mathtt{M}_F \end{bmatrix} \quad \mathtt{S} = \begin{bmatrix} \mathbf{S}_1 & \cdots & \mathbf{S}_P \end{bmatrix} \tag{2.3}$$

where $\mathtt{M}_i$ ($i = 1 \ldots F$) is the motion matrix relative to frame i, whose size depends on the camera model, and $\mathbf{S}_j$ ($j = 1 \ldots P$) encodes the 3D structure of point j, and its size depends on the kind of shape (for instance, rigid or non-rigid).

This decomposition was first observed and exploited by Tomasi and Kanade [110] to recover shape of a rigid scene in the case of orthographic projection. The factorisation algorithm proposed by Tomasi and Kanade [110] has been one of the most influential works in structure from motion. Introduced in the early 90's, it aims at recovering scene geometry and camera motion from an image sequence of a rigid object. Assuming a set of feature points are tracked through all the frames, a *measurement matrix* containing the image coordinates $(u_i, v_i)_f$ of every point i

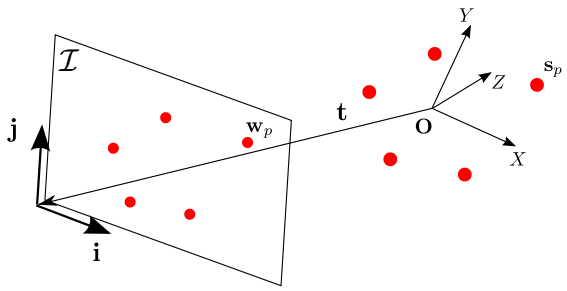


Figure 2.4: Projection of the rigid set of 3D points onto one image: vectors **i**, **j** and **t** encode camera rotation and translation with respect to object coordinate system with origin **O** in the object centroid.

for every frame f can be built. If there are P points tracked over F frames the $2F \times P$ matrix $\mathtt{W}$ can be expressed as equation 2.1.

Let $\mathbf{s_p} = (X_p, Y_p, Z_p)^T$ be the coordinates of a 3D point expressed in the world reference system. Assuming the orthographic projection model, image coordinates can be written as:

$$u_{fp} = \mathbf{i}_f^T(\mathbf{s}_p - \mathbf{t}_f) \qquad v_{fp} = \mathbf{j}_f^T(\mathbf{s}_p - \mathbf{t}_f) \tag{2.4}$$

where $\mathbf{i}_f$ and $\mathbf{j}_f$ are unit vectors pointing along the scan lines and the columns of the image in world coordinates, and $\mathbf{t}_f$ is the vector from the origin of the world coordinate system to the origin of the image plane at frame f, as illustrated in Figure 2.4. Vectors $\mathbf{i}_f$ and $\mathbf{j}_f$ are the first two rows of a 3×3 rotation matrix expressing camera rotation in the world coordinate system. Consider the matrix $\tilde{\mathtt{W}}$ obtained by subtracting the centroid of the image coordinates:

$$\tilde{u}_{fp} = u_{fp} - a_f \quad \tilde{v}_{fp} = v_{fp} - b_f$$

Where $a_f = \frac{1}{P}\sum_{p=1}^{P} u_{fp}$ and $b_f = \frac{1}{P}\sum_{p=1}^{P} v_{fp}$. The resulting matrix is called the *registered measurement matrix*. One important property of this matrix shown in [110] is the *rank theorem*, stating that under orthographic projection the rank of the registered measurement matrix of a set of tracked feature points is at most three. The proof of the rank theorem is straightforward, we report it here for its importance. The insight of Tomasi and Kanade's method is to centre

the world coordinate systems on the centroid of the 3D points. Recalling that 3D world coordinates are aligned with the centroid of the object, $\frac{1}{P}\sum_{q=1}^{P}\mathbf{s}_q = 0$. The projection equation for the registered measurement matrix can be written as:

$$\tilde{u}_{fp} = u_{fp} - a_f = \mathbf{i}_f^T(\mathbf{s}_p - \mathbf{t}_f) - \frac{1}{P}\sum_{q=1}^{P}\mathbf{i}_f^T(\mathbf{s}_q - \mathbf{t}_f) = \mathbf{i}_f^T(\mathbf{s}_p - \frac{1}{P}\sum_{q=1}^{P}\mathbf{s}_q) = \mathbf{i}_f^T\mathbf{s}_p$$

and similarly for $\tilde{v}_{fp}$, obtaining $\tilde{u}_{fp} = \mathbf{i}_f^T\mathbf{s}_p$ and $\tilde{v}_{fp} = \mathbf{j}_f^T\mathbf{s}_p$. These two sets of equations can be stacked in matrix form as:

$$\widetilde{\mathtt{W}} = \mathtt{RS}$$

where $\mathtt{R}$ and $\mathtt{S}$:

$$\mathtt{R} = \begin{bmatrix} \mathbf{i}_1^T \\ \mathbf{j}_1^T \\ \mathbf{i}_2^T \\ \mathbf{j}_2^T \\ \vdots \\ \mathbf{i}_F^T \\ \mathbf{j}_F^T \end{bmatrix} \qquad \mathtt{S} = \begin{bmatrix} \mathbf{s}_1 & \cdots & \mathbf{s}_P \end{bmatrix}$$

represent respectively the camera rotation and the shape estimate. Each matrix $\mathtt{R}_i$ is a 2×3 *truncated* rotation matrix, containing only the first two rows of the camera rotation matrix. The size of matrices $\mathtt{R}$ and $\mathtt{S}$ is $2F \times 3$ and $3 \times P$ respectively. Because the rank of these two matrices is at most three, the rank of their product must be at most three.

The result of the rank theorem is easily expressed in matrix form. Under an orthographic projection model, the location of feature points will be given by:

$$\mathtt{W} = \begin{bmatrix} \mathtt{R}_1 \\ \mathtt{R}_2 \\ \vdots \\ \mathtt{R}_F \end{bmatrix} \begin{bmatrix} \mathbf{s}_1 & \mathbf{s}_2 & \cdots & \mathbf{s}_P \end{bmatrix} + \mathbf{T}\mathbf{1}^T \tag{2.5}$$

Where $\mathtt{R}_i$ is the camera matrix for frame i, $\mathbf{s_p} = (X_p, Y_p, Z_p)^T$ is the vector of coordinates for a point p and $\mathbf{T}$ is the centroid of the 2D coordinates vertically stacked for all frames. The translation column-vector $\mathbf{T}$ is multiplied by a row-vector of ones to replicate the same translation vector on all columns.

Tomasi and Kanade show that it is possible to recover both factors $\mathtt{R}$ and $\mathtt{S}$ from the measurement matrix. The factorisation algorithm is based on the *singular value decomposition* (SVD) of the registered measurement matrix. SVD decomposes the matrix $\widetilde{\mathtt{W}}$ as:

$$\widetilde{\mathtt{W}} = \mathtt{U}\mathtt{D}\mathtt{V}^T$$

where $\mathtt{D}$ is the diagonal matrix of singular values, and $\mathtt{U}$ and $\mathtt{V}^T$ are the unitary matrices of singular vectors. Considering only the first three singular values, let $\mathtt{U}', \mathtt{D}', \mathtt{V}'^T$ be respectively the first three columns of $\mathtt{U}$, the first 3×3 minor of $\mathtt{D}$ and the first three rows of $\mathtt{V}^T$. The product $\mathtt{U}'\mathtt{D}'\mathtt{V}'^T$ minimises the Frobenius norm:

$$\begin{aligned} &\text{minimise} && ||\widetilde{\mathtt{W}} - \mathtt{W}'||_F \\ &\text{subject to} && \text{rank}(\mathtt{W}') = 3 \end{aligned} \tag{2.6}$$

The SVD thus gives an optimal rank-3 decomposition, let $\tilde{\mathtt{R}} = \mathtt{U}'\sqrt{\mathtt{D}'}$ and $\tilde{\mathtt{S}} = \sqrt{\mathtt{D}'}\mathtt{V}'^T$, this is called an affine decomposition. As we can see in Figure 2.6, the shape matrix does not represent an Euclidean reconstruction. This is because the Euclidean 3D shape (up to overall scale and rotation) will provide the observed feature tracks only when the direction vectors the image plane in $\mathtt{R}$ are orthogonal. Computing the correct camera matrices from affine ones is commonly

called a metric upgrade of the reconstruction. To obtain a metric upgrade, the key observation is that the decomposition is not unique, in fact, for any invertible matrix $\mathtt{Q}$:

$$\tilde{\mathtt{R}}\tilde{\mathtt{S}} = \tilde{\mathtt{R}}(\mathtt{Q}\mathtt{Q}^{-1})\tilde{\mathtt{S}} = (\tilde{\mathtt{R}}\mathtt{Q})(\mathtt{Q}^{-1}\tilde{\mathtt{S}}) = \mathtt{R}\mathtt{S}$$

It is possible to compute a matrix $\mathtt{Q}$ such that the rows of $\tilde{\mathtt{R}}\mathtt{Q}$ satisfy the orthonormality constraints:

$$\mathbf{i}_f^T \mathtt{Q}\mathtt{Q}^T \mathbf{i}_f = 1 \quad \mathbf{j}_f^T \mathtt{Q}\mathtt{Q}^T \mathbf{j}_f = 1 \quad \mathbf{i}_f^T \mathtt{Q}\mathtt{Q}^T \mathbf{j}_f = 0$$

This set of $3F$ equations encode the *metric constraints* that the matrix $\mathtt{R}$ must satisfy for the 3D structure and the camera matrices to live in Euclidean space.

The set of orthonormality constraints is a linear system of equations on the elements of the matrix $\mathtt{A} = \mathtt{Q}\mathtt{Q}^T$. Tomasi and Kanade's algorithm solves the metric upgrade problem by linearly computing $\mathtt{A}$ and then using Cholesky decomposition to obtain $\mathtt{Q}$. The Cholesky decomposition of a symmetric matrix $\mathtt{A}$ gives an upper triangular matrix $\mathtt{B}$ such that $\mathtt{A} = \mathtt{B}^T\mathtt{B}$. The product $\tilde{\mathtt{R}}\mathtt{Q}$ and $\mathtt{Q}^{-1}\tilde{\mathtt{S}}$ are respectively the updated motion and shape matrices, thus providing both camera pose and 3D structure in metric coordinates.

Example

Figure 2.7 shows the 3D reconstruction of the well known hotel sequence[1] Feature points were tracked using the **KLT**[2] tracker [68, 109]. Figure 2.5 shows some of the frames in this sequence. Figure 2.6 shows the 3D affine shape, before the metric upgrade. Figure 2.7 shows that the 3D reconstruction has been successfully upgraded to metric, as the walls of the house appear to be at right angles.

[1] http://vasc.ri.cmu.edu//idb/html/motion/hotel/index.html
[2] code available at http://www.ces.clemson.edu/~stb/klt/

Figure 2.5: Frames from the hotel sequence video

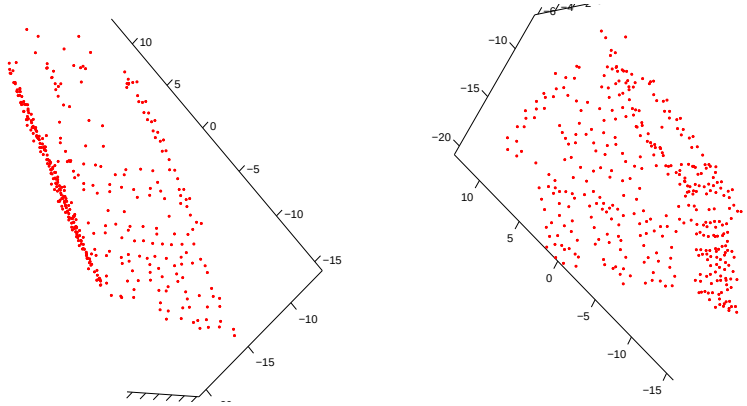


Figure 2.6: Example of Affine 3D reconstruction. The Shape matrix visualised before the metric upgrade step does not show a correct 3D structure.

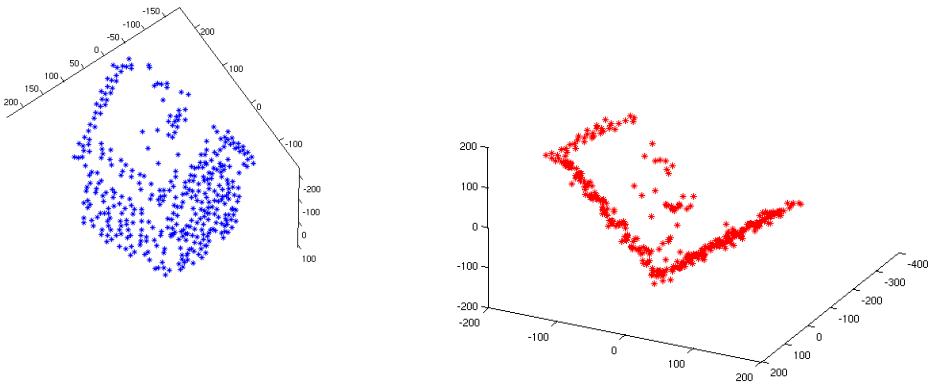


Figure 2.7: Left: 3D reconstruction of the hotel sequence, Right: Top view of the reconstructed 3D shape. Walls of the house are about at right angles, as expected

The seminal work of Tomasi and Kanade [110] introduced a solution for uncalibrated images, assuming an orthographic camera projection model. The algorithm was later extended to the case of multiple independently moving objects by Costeira and Kanade [28]. Kanatani and Sugaya in [59] analyse the computational complexity of rigid 3D reconstruction and provide algorithms for the weak-perspective and para-perspective projection models. Sturm and Triggs [104] proposed an extension of the factorisation algorithm to the case of a perspective camera. The next section will describe their method.

Perspective Factorisation

The orthographic projection model is an approximate camera model that works well if the relief of the object is small compared to its distance from the camera. The perspective projection model is a more accurate description of the image formation process. The projection of a 3D point $\mathbf{X}$ on the image plane under perspective projection is given by:

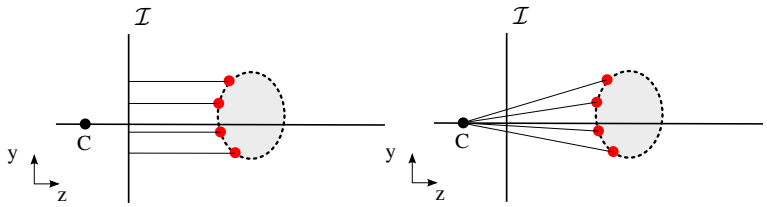
$$\mathbf{x} = \mathtt{P} \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix} \quad , \mathtt{where\ P} = \mathtt{K}[\mathtt{RT}] \tag{2.7}$$

The matrix $\mathtt{K}$ is the camera calibration matrix which encodes its intrinsic parameters: focal length f_x, f_y, principal point $(u,v)^T$ and skew α expressed in matrix form as:

$$\mathtt{K} = \begin{bmatrix} f_x & \alpha & u \\ 0 & f_y & v \\ 0 & 0 & 1 \end{bmatrix} \tag{2.8}$$

The rotation matrix $\mathtt{R}$ and the translation vector $\mathtt{T}$ align the *world* and *camera* reference frames and $\mathbf{x}$ is a 3×1 homogeneous vector such that the coordinates of the point on the image plane $\mathbf{w}_{ij} = (u_{ij}\ v_{ij})^T$ are given by its first and second elements divided by the third:

$$u = \frac{x_1}{x_3}\ v = \frac{x_2}{x_3} \tag{2.9}$$



(a) Orthographic projection (b) Perspective projection

Figure 2.8: a) Orthographic projection assumes rays from the object to the image plane **I** are parallel. b) Perspective projection takes into account that all rays intersect at the camera centre **C**.

The method proposed by Sturm and Triggs [104] for perspective reconstruction is based on the idea of defining and computing an additional unknown for each point, called the *projective depth*. Equations 2.7 and 2.9 imply:

$$\lambda_{ij}\bar{\mathbf{w}}_{ij} = \mathtt{P}_i \begin{bmatrix} \mathbf{X}_j \\ 1 \end{bmatrix} \tag{2.10}$$

Where $\mathtt{P}_i = \mathtt{K}_i[\mathtt{R}_i\mathtt{T}_i]$, and the image projection is expressed in homogeneous coordinates as $\bar{\mathbf{w}}_{ij} = (u_{ij}, v_{ij}, 1)^T$. λ_{ij} is an unknown projective depth for each point in each frame. Equation 2.10 can be expressed in matrix form for all points in all views as:

$$\bar{\mathtt{W}} = \begin{bmatrix} \lambda_{11}\bar{\mathbf{w}}_{11} & \dots & \lambda_{1P}\bar{\mathbf{w}}_{1P} \\ \vdots & \ddots & \vdots \\ \lambda_{F1}\bar{\mathbf{w}}_{F1} & \dots & \lambda_{FP}\bar{\mathbf{w}}_{FP} \end{bmatrix} = \begin{bmatrix} \mathtt{P}_1 \\ \mathtt{P}_2 \\ \vdots \\ \mathtt{P}_F \end{bmatrix} \mathtt{S} = \mathtt{MS} \tag{2.11}$$

with $\mathtt{S}$ the $4 \times P$ matrix of 3D points in homogeneous coordinates ($[\mathbf{X}_j^T 1]^T$). $\bar{\mathtt{W}}$ is called the rescaled measurement matrix. Because $\mathtt{M}$ and $\mathtt{S}$ have at most rank 4, such matrix is constrained to have *rank* ≤ 4. If all the projective depths were known, it would be possible to factorise $\bar{\mathtt{W}}$ into $\tilde{\mathtt{M}}\tilde{\mathtt{S}}$ using SVD. Such factorisation would give the 3D reconstruction up to a projective transformation. Since the projective depths are unknown, the main problem of perspective factorisation

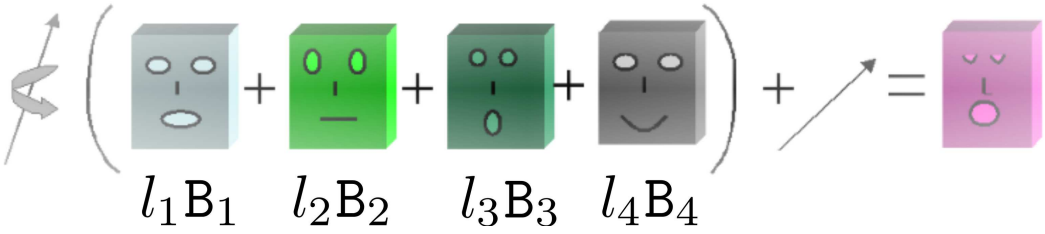


Figure 2.9: An illustration of the basis shape model for deformable objects: The shape in each frame in the sequence can be expressed as a linear combination of a set of fixed (but unknown) 3D basis shapes ($\mathtt{B}_1, \ldots, \mathtt{B}_K$) with time-varying coefficients ($l_1, \ldots, l_K$).

is computing the depths λ_{ij}. Sturm and Triggs [104] proposed the first method to extend the factorisation framework to the case of perspective projection. They solve for the projective depths by calculating the fundamental matrices and epipoles between pairs of views. The quality of the estimation depends strongly on the estimation of the fundamental matrices, which can suffer from image noise and poor initialisation. Iterative solutions have been proposed to improve convergence [117, 114, 55]. More recently, Dai *et al*. [29] proposed to globally compute the perspective weights by convex optimisation. To relax the non-convex constraint on the rank of the scaled measurement matrix, they minimise the nuclear norm instead, leading to a convex problem which approximates the low-rank constraints, thus obtaining a global solution.

2.3 Non-Rigid Structure from Motion

When the camera is viewing a non-rigid object, such as a moving human face talking or performing facial expressions, moving cloth, a flag waving in the wind or internal organs observed with an endoscope, its shape can change over time. The goal of Non-Rigid Structure from Motion (NRSfM) is to recover both the varying 3D shape of the object in each different frame, and the pose of camera given only a set of 2D image points, matched throughout the sequence. Since the shape of the object varies in time, the recovered 3D model should capture its deformations. In landmark work, Bregler *et al*. [15] were the first to demonstrate that it is possible under affine viewing conditions to infer the principal modes of deformation of a non-rigid object alongside its 3D shape within a structure from motion estimation framework. The key assumption is

that the 3D deformable shape can be represented as a linear combination of 3D basis shapes which encode the main modes of deformation — a so called *3D morphable model*. Figure 2.9 illustrates the basis shapes model. Their insight was that since this representation is linear it fits naturally into the factorisation framework. Once more, the underlying geometric constraints are expressed as a rank constraint which is used to factorise the measurement matrix to estimate the 3D pose, configuration coefficients and a pre-specified number of (unknown) 3D basis shapes. The problem of NRSfM can also be interpreted as an unsupervised learning problem in which the goal is to learn a low-rank 3D morphable model given only the 2D observations of the shape deforming over time.

2.3.1 Formulation

Given a video sequence of a deformable object, points on the 3D surface of the object are projected onto a set of 2D image trajectories by a moving camera. The object's deformability implies that the coordinates of the 3D points can change from frame to frame. As in the rigid case, the non-homogeneous coordinates ($\mathbf{w}_{fp} = (u_{fp}\ v_{fp})^T$) of P 2D image points observed in F frames can be collected in a measurement matrix:

$$\mathtt{W} = \begin{bmatrix} \mathbf{w}_{11} & \dots & \mathbf{w}_{1P} \\ \vdots & \ddots & \vdots \\ \mathbf{w}_{F1} & \dots & \mathbf{w}_{FP} \end{bmatrix} = \begin{bmatrix} \mathtt{W}_1 \\ \vdots \\ \mathtt{W}_F \end{bmatrix} \tag{2.12}$$

Assuming orthographic projection, and denoting $\mathtt{R}_f$ the 2×3 camera matrix for frame f, the 2D coordinates for all frames in all views are related to the varying 3D structure by:

$$\mathtt{W} = \begin{bmatrix} \mathtt{R}_1 & & \\ & \ddots & \\ & & \mathtt{R}_F \end{bmatrix} \begin{bmatrix} \mathtt{S}_1 \\ \vdots \\ \mathtt{S}_F \end{bmatrix} + \mathbf{T}\mathbf{1}^T \tag{2.13}$$

Where $\mathtt{S}_f$ is a $3 \times P$ matrix with the 3D coordinates of all P points in frame f, and $\mathbf{T}$ stacks the

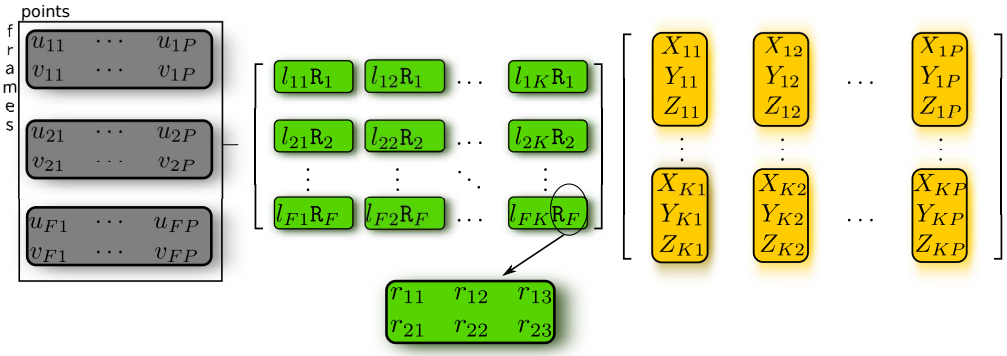


Figure 2.10: Formulation of the non-rigid factorisation problem. The measurement matrix can be decomposed into the product of a motion matrix that encodes the time-varying coefficients and the camera matrices, and a shape matrix containing the 3D basis vectors for all the points. The camera matrix is a 2×3 matrix with orthogonal rows. This formulation was introduced by Bregler *et al*. [15], whose method is detailed in Section 2.3.2.

camera translation vectors for all the frames. The goal of NRSfM is the joint estimation of the camera matrices and the deformable 3D structure. However, the non-rigid structure from motion problem is inherently under-constrained. It is clear that, if the 3D points move randomly, the problem is ill-posed, as the $3F \times P$ unknown 3D positions should be recovered from only $2F \times P$ data points. To resolve these inherent ambiguities, prior knowledge about either the shape of the object or the nature of the deformations must be used in formulating the problem.

The Low-Rank Basis Shape model

In 2000 Bregler *et al*. [15] were the first to observe that introducing statistical priors on the non-rigid 3D shape was enough to allow the non-rigid structure from motion problem to be solved within the popular factorisation framework. The 3D shape of a deformable object does not vary randomly over time, instead, the shape at each frame can often be expressed as a linear combination of a set of fixed (but unknown) basis shapes $\mathtt{B}_1, \cdots, \mathtt{B}_K$, weighed by time-varying coefficients $\mathbf{l}_f = [l_{f1}, l_{f2}, \cdots, l_{fK}]$ (one set of coefficients for each frame f). Figure 2.9 illustrates the low-rank basis shape model.

The set of 3D point coordinates that encodes the shape at each time frame f is given by:

$$\mathtt{S}_f = \sum_{d=1}^{K} l_{fd}\mathtt{B}_d \tag{2.14}$$

where $\mathtt{S}_f$ is the $3 \times P$ matrix that encodes the 3D coordinates of the P points on the surface of the object in frame f; $\mathtt{B}_d$ are the $3 \times P$ matrices encoding the shape basis and l_{fd} are scalars representing the deformation weights. Note that while the deformation coefficients vary from frame to frame to encode the non-rigid shape, the shape basis is fixed. It is precisely this low-rank representation of the shape that allows the joint estimation of non-rigid shape and camera motion within a factorisation framework.

With this shape model and assuming affine viewing conditions the projection equation for each frame becomes:

$$\mathtt{W}_f = \mathtt{R}_f(\sum_{d=1}^{K} l_{fd}\mathtt{B}_d) + \mathbf{T}_f \tag{2.15}$$

Similarly to the rigid case, the measurement matrix can be registered to the centroid of the 2D coordinates, such that the translation vector $\mathbf{T}_f$ becomes zero:

$$\tilde{\mathtt{W}}_f = \mathtt{R}_f(\sum_{d=1}^{K} l_{fd}\mathtt{B}_d)$$

We can now see that the NRSfM problem becomes the joint estimation of the camera matrices, deformation coefficients, and the shape basis. This is a tri-linear estimation problem where no prior information is assumed about the basis shapes; only the number of elements in the basis is known in advance.

2.3.2 Bregler *et al.*'s Original Non-Rigid Factorisation Algorithm

The work by Bregler, Herzmann and Biermann [15] was the first to extend the factorisation method to deformable objects. It was their insight of assuming the low-rank basis shape model, described in equation 2.14 in the previous section, that allowed to formulate non-rigid shape

estimation within the factorisation framework. Their work pioneered and established the new research area of Non-Rigid Structure from Motion in which this thesis is framed.

Similarly to the rigid case described in 2.2.3, the only input to the algorithm is the set of 2D coordinates of the image points tracked throughout the sequence. The original formulation assumes full data: all the points are visible in all the frames. The registered measurement matrix is rank deficient and can be factorised into the product of two low-rank matrices — the motion matrix $\mathtt{M}$ and the shape matrix $\mathtt{S}$:

$$\tilde{\mathtt{W}} = \begin{bmatrix} l_{11}\mathtt{R}_1 & \dots & l_{1k}\mathtt{R}_1 \\ \vdots & \ddots & \vdots \\ l_{F1}\mathtt{R}_F & \dots & l_{FK}\mathtt{R}_F \end{bmatrix} \begin{bmatrix} \mathtt{B}_1 \\ \vdots \\ \mathtt{B}_K \end{bmatrix} = \begin{bmatrix} \mathtt{M}_1 \\ \vdots \\ \mathtt{M}_F \end{bmatrix} \begin{bmatrix} \mathtt{B}_1 \\ \vdots \\ \mathtt{B}_K \end{bmatrix} = \mathtt{MS} \tag{2.16}$$

Where the matrices $\mathtt{B}_1, \cdots \mathtt{B}_K$ are the set of K 3D basis shapes, l_{fd} is the deformation coefficient (or configuration weight) that multiplies basis $\mathtt{B}_d$ in frame f and $\mathtt{R}_1, \cdots, \mathtt{R}_F$ are the 2×3 camera matrices for each frame. Equation 2.16 expresses in matrix form the orthographic projection of all the points on the non-rigid object in all the frames and shows that the registered measurement matrix $\tilde{\mathtt{W}}$ has at most rank $3K$, with K the number of deformation modes. Therefore, it can be factorised into the product $\tilde{\mathtt{W}} = \mathtt{MS}$ where the $2F \times 3K$ motion matrix $\mathtt{M}$ encapsulates all the time-varying parameters (deformation coefficients and camera matrices) and the $3K \times P$ shape matrix $\mathtt{S}$ encodes the 3D coordinates of P points on all the basis shapes.

However, this factorisation is not unique since any invertible $3K \times 3K$ matrix $\mathtt{Q}$ can be inserted in the decomposition leading to the alternative factorisation: $\mathtt{W} = (\tilde{\mathtt{M}}\mathtt{Q})(\mathtt{Q}^{-1}\tilde{\mathtt{S}}) = \mathtt{MS}$. The problem is to find the transformation matrix $\mathtt{Q}$ that imposes the appropriate replicated block structure on the motion matrix shown in equation 2.16 and that imposes the orthonormality constraints on the camera matrices $\mathtt{R}_i$ removing the affine ambiguity and upgrading the reconstruction to a metric one.

Bregler *et al.*'s non-rigid factorisation method follows exactly this two stage approach: first obtain an initial affine decomposition of the measurement matrix into two low-rank matrices via

singular value decomposition, followed by an upgrade step where the unknown linear transformation $\mathtt{Q}$ is estimated to impose metric constraints.

In [15] a linear approach to the computation of the metric upgrade transformation was proposed. The solution, named *sub-block factorisation*, is based on rearranging the elements of each sub-block of the motion matrix as

$$\mathtt{M}_f = [l_{f1}\mathtt{R}_f \dots l_{fK}\mathtt{R}_f] \tag{2.17}$$

Here $\mathtt{M}_f$ is the $2 \times 3K$ sub-block of the motion matrix related to frame f. Let r_i, $i = 1,\dots,6$ be the 6 elements of the camera matrix R_f, the block can be re-written as:

$$\mathtt{M}_f = \begin{bmatrix} l_{f1}r_1 & l_{f1}r_2 & l_{f1}r_3 & \dots & l_{fK}r_1 & l_{fK}r_2 & l_{fK}r_3 \\ l_{f1}r_4 & l_{f1}r_5 & l_{f1}r_6 & \dots & l_{fK}r_4 & l_{fK}r_5 & l_{fK}r_6 \end{bmatrix} \tag{2.18}$$

Rearranging the elements of $\mathtt{M}_f$, it is possible to decompose:

$$\breve{\mathtt{M}}_f = \begin{bmatrix} l_{f1}r_1 & l_{f1}r_2 & l_{f1}r_3 & l_{f1}r_4 & l_{f1}r_5 & l_{f1}r_6 \\ l_{f2}r_1 & l_{f2}r_2 & l_{f2}r_3 & l_{f2}r_4 & l_{f2}r_5 & l_{f2}r_6 \\ & & & \dots & & \\ l_{fK}r_1 & l_{fK}r_2 & l_{fK}r_3 & l_{fK}r_4 & l_{fK}r_5 & l_{fK}r_6 \end{bmatrix} = \begin{bmatrix} l_{f1} \\ l_{f2} \\ \dots \\ l_{fK} \end{bmatrix} \cdot [r_1 r_2 r_3 r_4 r_5 r_6] \tag{2.19}$$

thus proving that the values for the basis shape coefficients $\mathbf{l}_f$ for frame f could be recovered by a rank-1 factorisation of the rearranged motion matrix via singular value decomposition. Finally, since the rank-1 decomposition does not result in camera matrices $\mathtt{R}_f$ with orthonormal rows, orthonormality constraints must be enforced on the camera matrices as in [110] by solving a least-squares problem.

Bregler *et al.*'s original solution constitutes landmark work since it was the first to show that the factorisation approach can be applied to non-rigid objects. Moreover, the algorithm is attractive due to its simplicity and linearity. However, it suffers from various drawbacks. First the nested SVD approach is not robust to noise. When the sub-blocks $\mathtt{M}_f$ of the measurement matrix are

affected by noise, the rearranged matrix $\check{\mathtt{M}}_f$ will not be rank-1 and the further singular values will retain some of the contribution to the solution leading to errors. Second, the estimation of the upgrade transformation $\mathtt{Q}$ is only approximate. The estimated matrix is block diagonal, while the true metric upgrade matrix is dense in the off-diagonal values. As a consequence this method can only be used in the case of small deformations. Finally, the method assumes full complete point tracks: all the points must be seen in all the views which seriously hinders its application to real world scenarios.

Despite its drawbacks, Bregler *et al.*'s original non-rigid factorisation algorithm sparked enormous interest in the structure from motion community and soon new research followed which has progressively addressed many of the shortcomings of their approach.

2.3.3 An ill-posed problem

The recovery of the 3D structure of a deformable object from a sequence of images acquired with a single camera is an inherently ill-posed problem since different shapes can give rise to the same image measurements. In essence, the problem of reconstructing a non-rigid shape from an image sequence acquired with a single camera is equivalent to single-image reconstruction. Without the use of additional priors or constraints the problem is intractable given that the number of unknowns is higher than the amount of available data.

In the previous section we have described how adopting the simple but powerful prior that the deformable shape can be expressed with a linear subspace model allows to overcome some of the inherent ambiguities. However, ambiguities still remain in the non-rigid factorisation problem. The solution can only be computed up to an invertible transformation (the metric upgrade matrix) with additional scale ambiguities between the basis shapes and coefficients. Noise in the image measurements also causes the problem to be ill-conditioned as we saw was the case in Bregler *et al.*'s [15] original factorisation formulation. To overcome the inherent ambiguities, additional priors must be incorporated into the non-rigid structure from motion problem.

The rigidity of an object or scene has proved to be a sufficient constraint to enable to perform

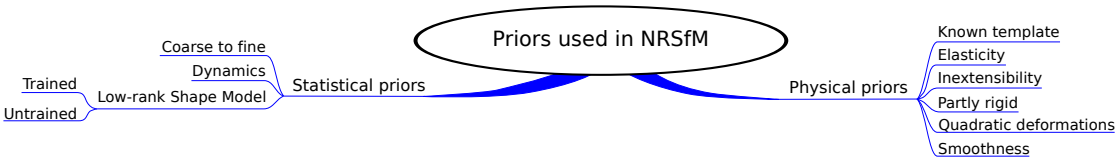


Figure 2.11: Use of priors in non-rigid shape estimation. The NRSfM problem is ill-posed. To overcome its inherent ambiguities, additional priors must be incorporated into the non-rigid structure from motion problem. We classify these priors into physical and statistical which are described in Section 2.3.3.

3D reconstruction from image sequences even in the case when no other prior information is available about the shape, the camera or its motion. Although in the case of non-rigid objects the priors are much weaker, they still exist and must be incorporated into the estimation to allow for unambiguous solutions to be obtained. In this section we provide examples of different types of additional priors that have been proposed in the literature to solve the problem of non-rigid shape reconstruction from monocular sequences. Additionally, throughout the rest of the chapter we will also refer to the specific priors or constraints used in each of the NRSfM methods we review. We classify them into *physical* priors and *statistical* priors.

Physical priors

Objects do not deform in an arbitrary, random way. There are physical forces that act on the object to constrain the way in which it moves. Different constraints on the nature of the deformations have been proposed in the literature.

Examples of *physical priors* that have been successfully applied to NRSfM range from weak ones such as the assumption that the camera is viewing a single surface ($\mathcal{C}_0$ continuity) used implicitly by most methods, that the surface in itself is spatially smooth ($\mathcal{C}_1$ continuity) [16] or that it deforms smoothly in time [1]; to stronger priors such as the surface is inextensible [97, 16], developable [89], partially rigid [35], piecewise planar [120] (or rigid [107] or quadratic [41, 95]) or even template-based methods that rely on a reference image in which the 3D shape of the object is known [97, 90, 16].

One of the priors most extensively used throughout NRSfM is the use of temporal smoothness

information. Introduced by Aanæs and Khal [1] in the context of Maximum A Posteriori (MAP) estimation with bundle-adjustment, this prior assumes that the 3D shape does not deform much from one frame to the next, and is usually referred to as temporal smoothness. This notion of temporal smoothness has since been adopted by many other methods [105, 36, 8, 111, 94, 84, 41, 95].

When more specific knowledge is available about the nature of the object being reconstructed, stronger priors can be used to disambiguate the solution. For instance, when recovering the 3D geometry of human facial expressions Del Bue *et al*. [35] imposed the constraint that some points on the face (e.g. points on the nose or the temples) move rigidly while others deform. The use of this partial rigidity constraint improves the accuracy of the metric upgrade step, consequently improving the estimation of camera pose and the 3D reconstruction of the shape. In [32] more complex priors on the shape of the object can be incorporated. When reconstructing the non-rigid face of a human, a prior rigid 3D model of a different person can be used as a soft constraint for the non-rigid reconstruction of the original subject.

The local spatial smoothness of non-rigid surfaces has also been used as a powerful constraint in NRSfM. It takes the form of a regularisation term imposing that neighbouring points on the surface must have similar coordinates in 3D space. This spatial smoothness prior, originally introduced by Torresani *et al*. [112], was then incorporated by many others [12, 35, 8] into their formulations.

Statistical priors

Statistical priors take advantage of the fact that deformations are not random and instead exploit the high correlation between the 3D trajectories of different points on the same non-rigid surface. The most successful statistical prior used in the NRSfM literature is of course Bregler *et al*.'s [15] assumption that the shape of a non-rigid object can be expressed in a compact way as a linear combination of an unknown low-rank shape basis. This simple regularisation on the shape allows to reduce the ambiguities to the metric upgrade transformation.

A common assumption in many factorisation methods based on the low-rank shape model is to

assume that the first basis is the dominant component [12, 1, 112, 35]. This is often achieved by constraining the shape in each frame to be close to the mean component of the shape basis, initialised using a rigid factorisation algorithm. This assumption implicitly assumes that the deformations are small deviations with respect to a strong rigid component.

Bartoli *et al*. [8] take the low-rank basis shape model a step further and propose a coarse-to-fine shape prior where new deformation modes are added iteratively to capture as much of the variance left unexplained by previous modes as possible. This prior imposes the natural assumption that the first basis encodes most of the motion and the rest of the bases express less and less important modes of variation. This is a much stronger statistical prior than the original low-rank shape model that does not make any assumptions on the individual modes of deformation. It is shown to avoid ambiguities since each basis is estimated independently in an incremental way so its estimation does not affect the previously computed ones.

Other examples of statistical priors include Torresani *et al*.'s [111] Gaussian distribution priors on the deformation weights. This prior effectively acts as a temporal smoothness constraint, since it models the fact that deformation parameters should be similar between consecutive frames. Torresani's formulation also allows to incorporate temporal linear dynamical models in object shape.

2.4 A Taxonomy of Non-Rigid Shape Estimation from Monocular Sequences

In this section we provide a taxonomy of solutions to the problem of non-rigid shape estimation from monocular sequences. We have classified approaches firstly based on the model adopted to represent the non-rigid shape. For a number of years the prevalent model in the literature has been the (untrained) **low-rank linear basis shape model** proposed by Bregler *et al*. [15] which allowed to extend factorisation approaches to the non-rigid shape domain. However, solving for the ambiguities — in particular, solving for the metric upgrade — inherent to NRSfM has proved a more challenging problem than initially anticipated. It has called for different optimisation techniques to compute solutions where the camera matrices satisfy the orthonormality

constraints and the 3D reconstructions are not affine but metric. We classify methods that use the linear low-rank shape model further according to the optimisation method used in the estimation. These range from **closed-form solutions** that impose the metric constraints *explicitly* by estimating directly the entries of the metric upgrade matrix, to **non-linear optimisation methods** such as **alternation** or **bundle-adjustment** that describe the problem directly in terms of the variables involved (camera matrices, basis shapes and deformation coefficients) and impose the metric constraints *implicitly* via parametrisation or projecting the solutions onto the correct manifold.

On the other hand, new **alternative shape models** have recently emerged in the literature that are beginning to address the limitations of the linear low-rank shape model by allowing to explain more complex deformations. These include piecewise planar, rigid or quadratic models; locally linear shape manifolds; sparse shape basis or DCT trajectory basis which allow for more complex deformations than those explained by the linear model.

We also describe non-rigid shape estimation methods for sequences acquired by a single camera that fall outside of the scope of NRSfM but are closely related. Shape estimation methods that use **training data** to learn the linear shape basis via principal components analysis (PCA) have been popular in the literature, particularly in the case of face modelling (active appearance models or morphable models). Here, the shape basis is known in advance and the shape estimation is limited to the tracking of the deformation coefficients and the camera matrix for each frame.

Template-based methods on the other hand, rely on a reference image in which the 3D shape of the observed object is known in advance. In principle, these methods work for pairs of images instead of a long image sequence: given an image and 2D-3D correspondences with a known 3D template, the problem involves estimating the deformed 3D shape in a new image. These template-based methods also require additional constraints to be imposed to avoid inherent ambiguities. We describe the practical solutions that have been proposed in the literature which include imposing physical priors on the shape such as its inextensibility.

Figure 2.12 illustrates this taxonomy: NRSfM methods are divided into those based on the

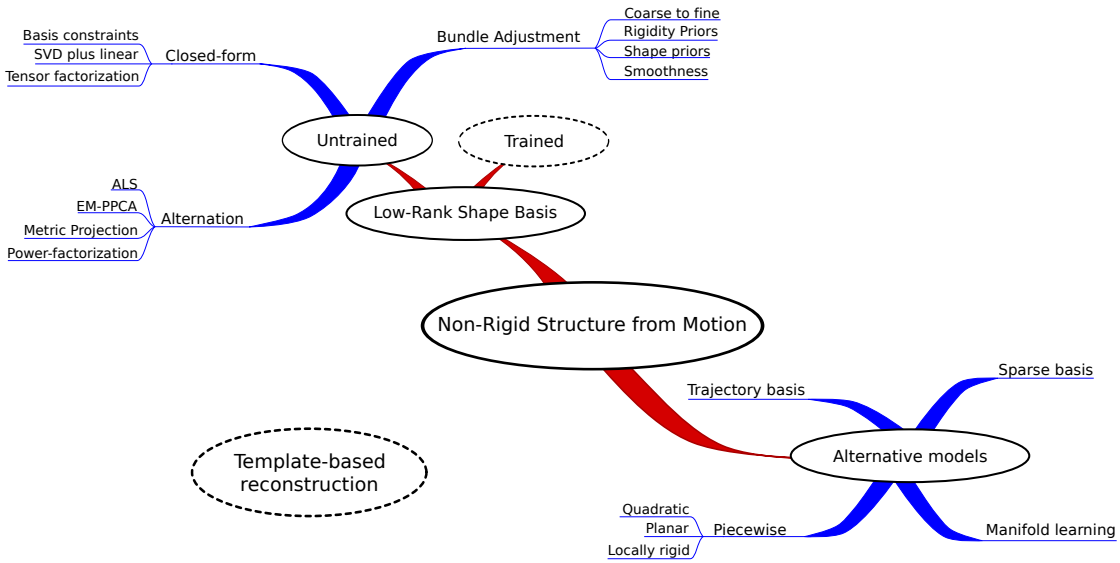


Figure 2.12: A taxonomy of proposed solutions to the Non-Rigid Structure from Motion problem. We divide NRSfM methods into those that use the low-rank basis shape model (untrained methods do not assume a known shape basis vs. trained ones that do) and those that use other shape models such as piecewise local models, locally linear shape manifolds, sparse shape basis or trajectory basis. Untrained methods based on the low-rank basis shape model are then classified according to the estimation method. The ambiguities inherent to the NRSfM problem have called for many different optimisation schemes to be proposed. We group these into closed-form solutions, alternation methods and non-linear least-squares optimisation or bundle-adjustment. Template-based methods for non-rigid shape reconstruction fall outside of the NRSfM framework since the input to the system is a 3D template and 2D-3D correspondences with a single image, instead of 2D correspondences throughout a long sequence. Strictly speaking, methods that use trained low-rank shape models also fall outside of the NRSfM since the shape model is known in advance.

low-rank linear basis shape model and those that use alternative shape models that allow more complex deformations. The first group are further divided according to the optimisation strategy into closed-form solutions and non-linear optimisation methods which include alternation and bundle-adjustment. Template-based methods and methods that use training data to learn the linear shape basis fall outside the NRSfM formulation but constitute popular alternatives in the literature to non-rigid shape estimation from monocular sequences.

2.5 NRSfM with the linear low-rank basis shape model

The assumption adopted by the original NRSfM formulation of Bregler *et al*. [15] that non-rigid shapes live in a low-dimensional linear subspace led to an estimation problem that fits perfectly within the powerful factorisation framework. Since then, many different methods have been proposed to solve the problem of factorising the measurement matrix into the product of two matrices that encode the non-rigid shape and the camera motion. However, as mentioned in section 2.3.3, the factorisation problem is subject to ambiguities. In particular, the decomposition can only be obtained up to an invertible transformation $\mathtt{Q}$. The estimation of this matrix $\mathtt{Q}$ that upgrades the reconstruction from affine to metric space has been the focus of NRSfM. In general, adding additional constraints leads to an optimisation problem that can be tackled in two different ways.

The first family of approaches involve estimating the elements of the metric upgrade matrix $\mathtt{Q}$ explicitly. Encouraged by the success of the factorisation algorithm in the case of rigid structure, researchers in NRSfM tackled the problem using an equivalent two-step approach: first an affine reconstruction is obtained factorising the measurement matrix via singular value decomposition imposing the rank constraint, followed by the direct estimation of the elements of the metric upgrade transformation matrix $\mathtt{Q}$. Bregler *et al*.'s method followed this exact approach [15] and many other so called closed form solutions have followed for the cases of both affine [128] and perspective [130, 49] viewing conditions. However, as noted by other authors, while they give an exact solution in the noise-free case, closed form solutions are known to break down in the presence of image noise [13, 111].

The second alternative is to write the non-linear optimisation problem in terms of the original variables that must be estimated: the camera matrices $\mathtt{R}_f$, the deformation coefficients l_{df} and the basis shapes $\mathtt{B}_d$ and the translations $\mathbf{t}_f$ if also optimised and minimise the squared reprojection error between the image locations of observed and predicted image points in all the views in

which they are visible which leads to the optimisation:

$$\underset{\mathtt{R}_f, l_{fd}, \mathtt{B}_d, \mathbf{t}_f}{\text{minimise}} \sum_{f,p} ||\mathbf{w}_{fp} - (\mathtt{R}_f \sum_{d=1}^{K} l_{fd}\mathtt{B}_d + \mathbf{t}_f)||^2 \tag{2.20}$$

This optimisation problem, does not involve the estimation of the upgrade matrix explicitly but instead solves for camera matrices that implicitly satisfy the metric constraint. This is normally achieved either via parametrisation or imposing constraints on the optimisation. Additional prior knowledge can be added to the cost function as regularisation terms or hard constraints.

2.6 Closed-form Solutions to NRSfM

Encouraged by the success of rigid factorisation algorithms, closed-form solutions to the NRSfM problem attempt to solve the problem following the same two step process: factorisation of the measurement matrix into the product of motion and shape matrices followed by explicit estimation of the metric upgrade matrix $\mathtt{Q}$. We will now describe the most influential approaches.

2.6.1 Basis constraints: Xiao-Chai-Kanade

The work of Xiao *et al*. [128] constituted a milestone in deformable structure recovery since they proposed an algorithm to recover the corrective transformation and solve the NRSfM problem in closed form both in the cases of orthographic and perspective cameras. Their work is of theoretical importance since they characterised the ambiguities present in NRSfM. Unfortunately, they concluded incorrectly that the orthonormality constraints alone were not enough to obtain an unambiguous solution to the NRSfM problem. However, their work greatly influenced and shaped the field.

The goal of closed form solutions is to estimate the invertible corrective transformation $\mathtt{Q}$ that yields the exact metric structure ($\mathtt{MS} = \tilde{\mathtt{M}}\mathtt{Q}\mathtt{Q}^{-1}\tilde{\mathtt{S}}$). Let $\mathtt{G} = \mathtt{Q}\mathtt{Q}^T$ be the corrective transformation multiplied by its transpose, each column triplet $\mathtt{Q}_k$ of $\mathtt{Q}$ is the transformation concerning each

single basis k.

$$\mathtt{M}^k = \tilde{\mathtt{M}}\mathtt{Q}_k$$

Where $\mathtt{M}^k$, $k = 1,...,K$ is a column triplet of the motion matrix $\mathtt{M}$. Recalling the structure of the motion matrix given in equation 2.16:

$$\mathtt{M}^k = \begin{bmatrix} l_{1k}\mathtt{R}_1 \\ l_{2k}\mathtt{R}_2 \\ \vdots \\ l_{Fk}\mathtt{R}_F \end{bmatrix} \tag{2.21}$$

The column block $\mathtt{M}^k$ contains the camera matrices for all frames, scaled by the corresponding coefficient of the k^{th} basis shape l_{fk} for that frame (with $k = 1,...K$). Therefore the affine motion matrix $\tilde{\mathtt{M}}$ must have the form $\mathtt{M} = \tilde{\mathtt{M}}\mathtt{Q}$ such that:

$$\tilde{\mathtt{M}}_f \mathtt{G}_k \tilde{\mathtt{M}}_e^T = l_{fk} l_{ek} \mathtt{R}_f \mathtt{R}_e^T \tag{2.22}$$

Where $\mathtt{G}_k = \mathtt{Q}_k \mathtt{Q}_k^T$ is the $3K \times 3K$ transformation relative to the $\mathtt{Q}_k$ column triplet, and $\tilde{\mathtt{M}}_i$ are the two rows of the motion matrix relative to the i^{th} frame. Due to the orthonormality of the rows of the projection matrix for each frame f the *metric constraint* can be expressed as:

$$\tilde{\mathtt{M}}_f \mathtt{G}_k \tilde{\mathtt{M}}_f^T = l_{fk}^2 \mathtt{I}_{2x2} \tag{2.23}$$

where $\mathtt{I}_{2x2}$ is a 2×2 identity matrix. The diagonal elements yield a single equation since l_{fk} is unknown while the off-diagonal constraints are identical since $\mathtt{G}_k$ is symmetric. Therefore, for F frames, $2F$ constraints are obtained:

$$\tilde{\mathtt{M}}_{2f-1} \mathtt{G}_k \tilde{\mathtt{M}}_{2f-1}^T - \tilde{\mathtt{M}}_{2f} \mathtt{G}_k \tilde{\mathtt{M}}_{2f}^T = 0 \tag{2.24}$$

$$\tilde{\mathtt{M}}_{2f-1} \mathtt{G}_k \tilde{\mathtt{M}}_{2f}^T = 0 \tag{2.25}$$

Given a sufficient number of frames, the metric constraints should be enough to determine the entries of $\mathtt{G}_k$. However, Xiao *et al*. 's contribution was to provide a proof that the solution of equations 2.24 and 2.25 is ambiguous. They concluded (incorrectly) that orthonormality constraints are not sufficient on their own to solve for the upgrade matrix unambiguously. Omitting the details of the proof, they show that any solution to equations 2.24 and 2.24 (the metric constraints) has the form: $\mathtt{Q}\mathtt{H}_k\mathtt{Q}^T$, where $\mathtt{Q}$ is the desired transformation matrix, and $\mathtt{H}_k$ is a matrix given by the sum of an arbitrary block-skew-symmetric and an arbitrary block-scaled-identity matrix. This result means that for deformable shapes, the solution given by imposing the orthonormality constraints is ambiguous. In other words, the space defined by orthonormality constraints alone contains both correct and invalid solutions.

Since orthonormality constraints were considered not to be sufficient, to eliminate the ambiguity Xiao *et al*. proposed to introduce a set of novel constraints known as *basis constraints* which uniquely determine the shape basis resolving the ambiguity. They then proved that the orthonormality and basis constraints together led to a closed-form solution of the NRSfM problem. While their method recovers the ground truth solution in synthetic experiments, it has been shown to deteriorate quickly even for low levels of measurement noise and to be very sensitive to the choice of basis constraints. Moreover, it cannot be extended to deal with outliers or missing data in the tracking.

In defence of orthonormality constraints

As was suggested by Brand [13] and later proved by Akhter *et al*. [5], in the case of noise-free observations, orthonormality constraints are in fact sufficient to solve for the corrective transformation of NRSfM. Xiao *et al*.'s proof that orthonormality constraints were not sufficient to solve for the upgrade matrix was incomplete. Akhter *et al*. [5] showed that the reason for the unsolved ambiguities in [128] was that the rank of matrix $\mathtt{G}_k$ was not constrained to be 3. Imposing this additional constraint is sufficient to eliminate the ambiguities in the 3D reconstructed shape. However, the constraints are non-linear and very hard to optimise. Therefore, while orthonormality constraints were shown to be sufficient to resolve ambiguities, solving the

exact constraints involves a non-linear optimisation problem which can lead to undesirable local minima. In practice, Akhter *et al*. 's work did not provide a new algorithm. As we will see in Section 2.7 many different non-linear optimisation algorithms have been proposed to tackle this problem, either via parametrisation or imposing hard constraints.

2.6.2 Closed form solution for perspective cameras: Hartley-Vidal

Recently Hartley and Vidal [49] proposed a linear, closed form solution to the problem of recovering deformable structure when the perspective effects in the images cannot be ignored. This algorithm requires the initial estimation of a multi-focal tensor, based on their previous work in [48]. The tensor is then factorised into the projection matrices and then linear algebra techniques are used to enforce constraints on the projection matrices to estimate explicitly the corrective transformation from which the camera matrices, basis shapes and shape coefficients are computed. Although the entire approach is linear, the authors report that the initial tensor estimation and factorisation is very sensitive to noise.

2.6.3 Brand's direct method

In influential work by Brand [12] the metric upgrade estimation is guided by the assumption that the average mean shape should explain most of the image motion, leaving deformation components to model only the residual non-rigid motion. In later work [13] Brand estimates the invertible corrective transformation $\mathtt{Q}$ directly in terms of its gram matrix $\mathtt{Q}\mathtt{Q}^T$. As we described in section 2.6.1 this method avoids the ambiguities highlighted by Xiao *et al*. [128] and is the first to hint that orthonormality constraints are sufficient to perform metric upgrade.

Brand takes advantage of Kronecker product properties by writing the factorisation equation as:

$$\mathtt{W} = \mathtt{M}\mathtt{S} = \begin{bmatrix} \mathbf{I}_1^T \otimes \mathtt{R}_1 \\ \vdots \\ \mathbf{I}_F^T \otimes \mathtt{R}_F \end{bmatrix} \begin{bmatrix} \mathtt{S}_1 \\ \vdots \\ \mathtt{S}_K \end{bmatrix} \tag{2.26}$$

With $\mathbf{l}_f$ a vector of coefficients for each frame, $\mathtt{R}_f$ the camera matrix, and $\mathtt{S}_1, \cdots, \mathtt{S}_K$ the set of basis shapes. Brand's key contribution is to show that it suffices to determine a single column-triad $\mathtt{Q}_{1:3}$ of the corrective transformation to uniquely recover the entire solution. In addition to this property, this work also presents a new method to rewrite the metric constraints as the minimisation of a cost function with the geometric meaning of minimising the deviation from orthogonality in the resulting motion matrix. The minimisation is carried out using a quasi-Newton method with a closed-form optimum jump. As the paper points out, this global optimum is not guaranteed in the presence of noise in the measurement matrix.

It is important to note that although the algorithm recovers the ground truth solution on synthetic experiments its main weakness is its inability to handle noisy data. Nevertheless, Brand's method performs better than the closed form solution by Xiao *et al*. on real video sequences.

2.7 NRSfM via non-linear optimisation

We now review methods that solve the NRSfM problem minimising a non-linear geometric cost (image reprojection error) expressed in terms of the original variables: the camera matrices $\mathtt{R}_f$, the deformation coefficients l_{df} and the basis shapes $\mathtt{B}_d$. This amounts to a tri-linear estimation problem where the orthonormality constraints are usually imposed via parametrisation or imposing hard constraints. We describe optimisation methods based on alternate least-squares and bundle adjustment.

2.7.1 Alternation methods

Alternation is an iterative scheme that involves alternatively optimising each of the variables: rotations, shape basis, and shape coefficients, while keeping the others fixed. Since our Metric Projections algorithm (see Chapter 3) belongs to this category, in this section we describe other alternation based approaches and discuss the main differences between them.

Torresani et al.*'s Alternating Least Squares*

The first solution to the non-rigid structure from motion problem optimising image reprojection error via an alternating least-squares approach was proposed by Torresani *et al.* in [112]. In this work, the authors argue that an initial estimate of the camera matrices can be obtained accurately by using Tomasi and Kanade's rigid factorisation algorithm. Under this assumption, the recovery of the deformation coefficients, shape basis and camera matrices is recast as an alternating least-squares estimation: each of the three unknowns is iteratively estimated in turn while keeping the others constant. At each step of the iteration, the estimation of the shape basis and deformation weights was solved in closed-form, but the camera matrices are subject to a non-linear orthonormality constraint which cannot be updated in closed-form. Instead, a single Gauss-Newton step is performed which results in an approximation of the updated value of the camera matrices. The orthonormality of the rotation matrices is guaranteed by parameterising the incremental update in exponential map coordinates. This method relies on an accurate initial estimate, and requires an initialisation for the deformation weights, which were initialised with small random values in [112].

While they do indeed enforce the exact metric constraints through the exponential map parametrisation of the rotation matrices, unfortunately, the update of the camera matrix is only an approximation. Since their approximate camera update step, where orthonormality constraints are imposed, is closely related to our Metric Projections algorithm, we discuss it in detail in the next section.

Camera Matrix Update

In the alternation scheme, the 2×3 camera matrices $\mathtt{R}_f$ cannot be updated in closed form because of the nonlinear orthonormality constraint. Torresani *et al.* propose to parametrise the current estimate of $\mathtt{R}_f$ with a 3×3 rotation matrix $\mathtt{Q}_f$ as $\mathtt{R}_f = \Pi \mathtt{Q}_f$, where

$$\Pi = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \tag{2.27}$$

where Π selects the first two rows of a 3×3 rotation matrix $\mathtt{Q}_f$ to give the 2×3 orthographic camera matrix $\mathtt{R}_f$. The updated rotation $\mathtt{Q}_t^{new}$, relative to the previous estimate, is expressed in terms of an incremental rotation $\delta_{\mathtt{Q}_f}$ as $\mathtt{Q}_t^{new} = \delta_{\mathtt{Q}_f}\mathtt{Q}_f$.

The incremental rotation $\delta_{\mathtt{Q}_f}$ is parametrised in exponential map coordinates by a 3-vector $\xi_f = [w_f^x, w_f^y, w_f^z]$

$$\delta_{\mathtt{Q}_f} = e^{\hat{\xi_f}} = \mathbf{I} + \hat{\xi_f} + \hat{\xi_f}^2/2! + \cdots \tag{2.28}$$

Where $\hat{\xi_f}$ is the skew-symmetric matrix built from the vector ξ_f.

$$\hat{\xi_f} = \begin{bmatrix} 0 & -w_f^z & w_f^y \\ w_f^z & 0 & -w_f^x \\ -w_y^x & w_f^x & 0 \end{bmatrix}$$

Their strategy is then to estimate the parameters ξ by approximating $\mathtt{Q}_t^{new} = (\mathbf{I} + \hat{\xi}_t)\mathtt{Q}_t$ in the geometric cost function, then update $\mathtt{Q}_t^{new}$ as $e^{\hat{\xi}_t}\mathtt{Q}_t$. This approximate update of the rotation matrix is based on the idea that from one iteration to the next the rotation must only have small changes. Thus the initial values for the camera parameters must be close to the final solution for this assumption to be valid.

Torresani et al.*'s Probabilistic PCA model*

An influential approach to NRSfM was later proposed by the same authors [111]. The key idea was to replace the linear subspace shape model with a probabilistic PCA model introducing Gaussian priors on the deformation coefficients (i.e. assuming that they can be drawn from a Gaussian distribution), with the rest of the model defined as before. This prior represents an explicit assumption that the deformation coefficients for each pose will be similar to each other, that is, they are not unconstrained and the mean rigid component will explain most of the motion. The additional advantage of using this model is that the deformation weights become *latent variables* which are therefore not explicitly solved for but can be marginalised out. This results in a simpler optimisation problem with fewer variables.

With this model, NRSfM can be formulated as maximising the joint likelihood of the image measurements. This optimisation problem was then solved via Expectation Maximisation where all the model parameters are updated in closed form except for the camera rotation matrices. While they do enforce the exact metric constraints through an exponential map parametrisation of the rotation matrices, the update of the camera matrix is only an approximation — the camera matrix cannot be updated in closed form and instead they perform a single Gauss-Newton step. In practice, the same approximate rotation matrix update process is used as the one we described in section 2.7.1.

The model is then extended to include a linear dynamical model of the shape (LDS) in the probabilistic framework. The shape coefficients in a frame are modelled as a linear function of those in the previous one. Due to their ability to handle missing data and their resilience to measurement noise, the EM-PPCA and EM-LDS algorithms proposed by Torresani *et al*. [111] have become a standard benchmark in NRSfM. However, in practice, they can only deal with relatively simple deformations (small deviations from a mean rigid component) and for more complex deformations they have been outperformed by more recent approaches based on piecewise or local shape models [41, 95, 107] Moreover, other approaches such as our Metric Projections algorithm [86] (see Chapter 3) have been shown to cope with larger amounts of missing data.

Rotation-Constrained Power Factorisation: RCPF

A variation of Torresani *et al*.'s trilinear ALS alternation method [112] for the recovery of non-rigid structure was later proposed by Wang *et al*. in [123]. The idea is to use an ALS scheme equivalent to Torresani *et al*.'s [112], alternating between the estimation of each of the three factors (rotations, coefficients, and basis shapes) of the non-rigid estimation problem, assuming the other two known. The novelty is to include a projection step of every rotation matrix onto the Stiefel manifold (the set of matrices with orthonormal columns). In this respect, RCPF is related to our Metric Projections algorithm [86] described in Chapter 3, since it also includes a projection step of the rotation matrices onto the Stiefel manifold to impose the orthonormality constraints. However, the projection steps differ: while our Metric Projections approach projects

the entire motion matrix $\mathtt{M}$ onto the non-rigid motion manifold, RCPF only projects the rotation matrices onto the manifold of matrices with orthonormal columns. Our comparative results in Chapter 3 show that our Metric Projections algorithm [86] outperforms RCPF [123], particularly for high percentages of missing data.

2.7.2 Non-rigid bundle adjustment

Solving the non-rigid structure from motion problem requires the simultaneous estimation of the camera pose and deformation coefficients for every frame and the 3D coordinates of the basis shapes. Even in the case of a simple NRSfM problem with a couple of hundred points on a surface deforming over a short hundred frame sequence with ten basis shapes this results in a very large number of parameters to estimate (10^4). Direct non-linear minimisation of the image reprojection error over all the parameters is computationally expensive. This is a case where the bundle adjustment algorithm (described in section 2.2.2) is particularly useful: the computational cost is reduced greatly taking into account that each parameter interacts only with a few data points. The Levenberg-Marquardt algorithm [93] can be used to minimise the non-linear cost function, taking advantage of the sparse nature of the Jacobian.

For the non-rigid case, the cost function to be minimised is the reprojection error:

$$\sum_{f,p} ||\mathbf{w}_{fp} - (\mathtt{R}_f \sum_{d=1}^{K} l_{fd} \mathtt{B}_d + \mathbf{t}_f)||^2 \tag{2.29}$$

That is, the sum over all points p and all frames f, of the residual between the measured 2D feature location $\mathbf{w}_{fp}$, and the 2D position predicted by the model for that feature. $\mathtt{R}_f$, l_{fd}, $\mathtt{B}_d$, and $\mathbf{t}_f$ are the model parameters, respectively, the camera matrices, deformation weights, basis shapes, and 2D translations. Because of the difficulty in imposing orthonormality between the two rows of $\mathtt{R}$, the camera matrix is usually parametrised as a truncated rotation matrix, and the rotations parametrised with quaternions (or other parametrisation which ensure orthonormality). Additional regularisation priors are normally added to the cost function to give a Maximum A Posteriori estimate (MAP). These include spatial and temporal smoothness priors as we dis-

cussed in section 2.3.3. In general, the camera matrices and the mean shape are initialised with a rigid factorisation algorithm and the basis shapes and deformation coefficients are initialised to small random values.

We will now review the most successful methods that apply this optimisation technique to the non-rigid structure from motion problem.

Aanæs and Kahl non-rigid bundle adjustment

Aanæs and Khal [1] were the first to propose the joint non-linear optimisation of the shape and motion parameters minimising image reprojection error. They describe the ambiguities that arise from the increased number of degrees of freedom in non-rigid SfM as opposed to the rigid case and argue the need of priors to constrain the solution. In addition to showing that the non-rigid structure from motion problem can be solved via non-linear optimisation method, they introduce a temporal smoothness prior: the 3D shape should change little from frame to frame. This prior is implemented by adding an extra term to the cost function to penalise large changes in the deformation parameters from one frame to the next. The cost function included an additional prior that the shape should be close to an initial estimate computed using rigid factorisation techniques. They also were first to use the Bayesian Information Criterion (BIC) [74] model selection technique for the choice of the number of deformation modes.

Non-rigid bundle adjustment for a perspective camera using rigidity shape priors

The work by Del Bue *et al*. [31, 34] solves the problem of non-rigid shape reconstruction using a perspective camera using a bundle adjustment approach and adding priors on the degree of deformation of some of the points in the scene. Their work exploits the fact that it is often a reasonable assumption that some of the points are deforming throughout the sequence while others remain rigid. The set of rigid points is used to estimate the internal camera calibration parameters and the overall rigid motion. Finally the problem of non-rigid shape estimation is formalised as a constrained non-linear minimisation adding priors on the degree of deformability of each point. This method was extended to the case of perspective projection with varying intrinsic

parameters [64], and to the case of a stereo camera setup [33]. A further contribution proposed in [32] by Del Bue is the use of shape priors: it is shown that incorporating the knowledge of a previously computed 3D shape can improve the estimation of motion and deformations.

Coarse-to-fine estimation

Bartoli *et al*. [8] propose a way to avoid ambiguities in the estimation of the basis shapes by recovering them in an incremental way, one at a time, adding new deformation modes iteratively to capture as much of the variance left unexplained by previous modes as possible. An important characteristic of this method is the automated selection of the number of basis shapes, which usually has to be specified *a priori*, using cross-validation. In addition to the implicit prior of having each deformation mode express smaller and smaller deformations, two additional priors are imposed: temporal smoothness and spatial smoothness, which are shown to greatly improve the results [8]. The method relies on Bundle Adjustment to minimise the image reprojection error incorporating the priors.

2.8 Trained models for non-rigid shape analysis

When training data is available, learning deformable models using statistical learning methods has become an attractive way to represent non-rigid shape. In particular, learning lower-dimensional linear models from training data using PCA analysis has prevailed as a popular alternative in the literature, and has been applied with huge success to modelling of faces. The original 2D Active Shape Models (ASM) by Cootes and Taylor [24], were extended to include texture in the Active Appearance Models (AAM) [25]. AAM have been extensively applied in the literature to track 2D face deformations [71]. This class of methods relies on the availability of labelled information for a subset of images or *training data*. The model is separated into shape and texture components, both modelled as linear combinations of basis vectors learnt via Principal Components Analysis. The basis acquired in the training phase can be used to generate new instances of the model (for instance, a new pose for the face in the example of figure

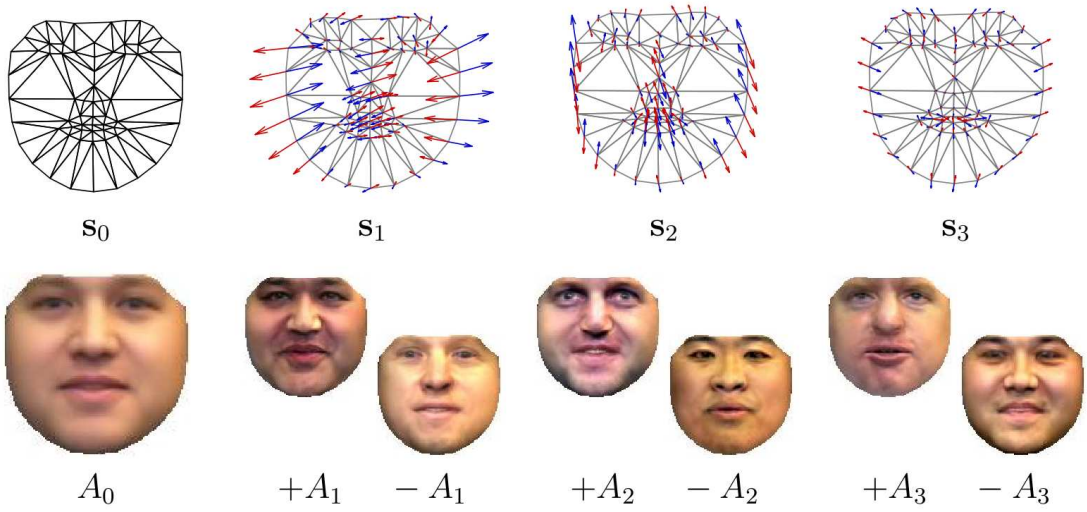


Figure 2.13: Example of a linear Active Appearance Model from [72]. Top Row: The average shape $\mathbf{s}_0$ is added to a linear combination of basis vectors $\mathbf{s}_1, \cdots, \mathbf{s}_3$ learnt from training data. Bottom row: Texture is generated by adding the average appearance $\mathtt{A}_0$ for each point to the learnt modes of appearance $\mathtt{A}_1, \cdots, \mathtt{A}_3$. Images courtesy of Iain Matthews and Simon Baker

2.13). The recovery of model parameters given a target image is then recast as the estimation of the best parameters that would generate the image given the known model. 2D Active Appearance Models were then extended to *3D morphable models* [121] to obtain full 3D models of faces, which produced impressive results given a single high-resolution image as input. The morphable model is computed from depth estimates from 3D scanning devices. It was later extended to model different facial expressions of the same face.

Learnt non-linear models have also been used to model non-rigid or articulated data. For instance GPLVM (Gaussian Process Latent Variable Models) have been successfully used for 3D people tracking to learn a prior on human pose [119] from small training data sets. In NRSfM, Salzmann *et al*. [99] proposed to learn a prior over the local deformation of surface patches from motion capture data. These local models were learnt with GPLVM from a small number of samples. The local to global transition was then made using a Product of Experts paradigm that combines probabilistic models.

While learnt models have proved very effective at representing non-rigid deformations they suf-

fer from two main drawbacks. First, gathering a sufficient amount of training data can be a difficult process. Secondly, alignment and labelling of the training data can be extremely time consuming and error prone since it is usually done manually. Recently some techniques have been proposed to automate as much as possible the process of obtaining the training data needed to build the model. For example Davies *et al*. [30] propose to automatically select model points along a contour of the objects to be modelled by minimising description length. Cootes *et al*. [27] construct a model from an initial position of feature points in the images, that can be obtained automatically starting with a regular grid, removing points on low texture areas and moving points to strong edges in the image. Deformations are modelled using an affine piece-wise deformation field, and training is performed using a Minimum Description Length approach.

Current state of the art methods include the work by Cootes *et al*. [26] and Adeshina and Cootes [2]. The former is suitable for automatic or semi-automatic operation, where the user has to label only one reference frame while the latter takes advantage of specific subject knowledge by asking the user to label a *parts and geometry* model in a reference frame, providing both appearance and spatial relationships of features, then learning the correspondences across images.

2.9 Reconstruction with Missing Data

The original formulation of the factorisation problem described in section 2.2.3 requires all the points to be visible in all the views. However this is often not a realistic assumption for tracking algorithms. The main sources of missing data are occlusions, self-occlusions and tracking failures (broken tracks). A common occurrence in real world scenes are occlusions: when the object that is being reconstructed is not entirely visible in the whole image sequence. For example, the hands of a person talking often move behind or in front of the body, sometimes occluding one another in the image. A cloth could fold, so that part of its surface is not visible, this is called a ”self-occlusion”. Also, the object and/or the camera will rotate, hence features in the front of the surface will disappear from view, and features at the back will appear. Finally, the tracking algorithm might fail to detect a feature in a particular frame, or lose track of a feature. All of these

kinds of occlusions result in the problem of missing information in the measurement matrix. This problem can be tackled at the feature detection and tracking stage, but most importantly we have to develop structure-from-motion techniques that will be able to produce accurate 3D reconstructions without full data.

The problem of missing data can thus be defined as the problem of computing an accurate model from incomplete observations. Although the problem of factorising a matrix into the product of two low rank factors in the presence of missing data has received great attention from researchers in computer vision, it continues to be an open problem that affects also other areas. Stemming from Wiberg's original algorithm [125], many different solutions have been since put forward for the structure from motion problem, when missing data arises due to occlusions or broken tracks. Being able to deal with high percentages of missing data is crucial for algorithms to be practical in real, not just controlled, scenarios. We will review the main approaches to solve the missing data problem.

One group of approaches propose strategies that combine partial low-rank factorisations obtained for sub-blocks of the measurement matrix that contain full data. Pioneered by Jacobs [58], these *batch approaches*, reconstruct the measurement matrix by first building its row or column null-space or one of its range spaces and have been applied both to the rigid [58, 106] and non-rigid [82] SfM problems. However, one concern about these approaches is their sub-optimality and that their performance degrades in the presence of noise.

A second group of missing data approaches that dominates the literature includes iterative methods that perform alternation of closed form solutions to solve for the two factors of the matrix [21, 46, 47, 112]. For instance, Powerfactorization [47] uses an alternated least-squares (ALS) scheme to solve for the motion and shape matrices. Alternation constitutes an attractive scheme since it guarantees convergence to a local minimum as the objective function is reduced after each iteration. Further advantages are quick iteration steps combined with a fast convergence rate in the initial iterations. However, after a few iterations, the convergence rate drops and these algorithms are susceptible to flat lining.

Finally, non-linear optimisation algorithms have also been proposed to optimise directly the reprojection error. Buchanan and Fitzgibbon [17] proposed a Damped Newton algorithm that provides a more robust solution than standard alternation approaches. Later, Chen revitalised the use of the Levenberg-Marquardt algorithm to solve the missing data problem by formulating the low-rank matrix approximation problem as a minimisation on subspaces [20]. The idea is to consider the shape as an implicit function of the motion and measurement matrices and solve only for the motion. This results in a smaller system to be solved in every iteration, which makes this method well suited to large matrices where it outperforms Wiberg's [125] method or Damped Newton [17]. Recently Candès and Recht have proposed a convex optimisation method [18].

Although these non-linear methods do exhibit a superior performance, proper initialisation remains an open problem and, more importantly, integrating additional constraints in the optimisation process is not an easy task.

2.10 Alternative shape models

Despite the success of the linear low-rank basis shape model in the NRSfM literature, in recent years new research has began to address its main limitation of only being able to model small deviations with respect to a strong rigid component. In order to allow more complex deformations, authors have departed from the linear basis shape model and proposed new shape models that can cope with a wider range of non-rigid motion. In particular, we will review approaches based on *piecewise* models, *locally-linear shape manifolds*, and *trajectory space* basis.

2.10.1 Piecewise reconstruction methods

Following Bregler *et al.*'s [15] original non-rigid factorisation algorithm, most NRSfM algorithms represent the time varying 3D shape as a linear combination of a low rank shape basis. Although this model effectively captures global deformations, and many approaches have been proposed [8, 12, 35, 111, 128], so far, they have only been demonstrated on simple sequences

where the deformations are small linear deviations from a mean rigid component and none of them are able to reconstruct strongly deforming surfaces such as a piece of cloth deforming vigorously. This failure can be attributed to their reliance on a global model — to capture intricate local deformations, global models require a substantial increase in the number of basis shapes used which leads to over-fitting.

Recently, *piecewise* reconstruction methods have been proposed in the NRSfM literature as an alternative to global ones. Their insight is that models that attempt to capture the scene's global spatio-temporal behaviour — such as the low-dimensional linear shape subspace favoured by most non-rigid SfM methods — are unable to handle complex deformations often leading to over-fitting. Instead, they decompose the global reconstruction problem into many better-behaved local ones relying on the features shared between overlapping patches to enforce global consistency. Local solutions are *stitched* into a global surface imposing the constraint that the points shared between patches are the same points in 3D space.

The first of such approaches was proposed by Varol *et al*. [120] assuming that the 3D surface can be approximated as piecewise planar. The method works on image pairs and assumes a calibrated camera. The image is first divided manually into overlapping regions which are reconstructed independently as local 3D planes from the homographies estimated from pairwise correspondences. The patches are then merged by enforcing 3D consistency between the overlapping points on neighbouring patches. In a final step, a triangular mesh is fit to the 3D point cloud assuming temporal smoothness constraints. The strength of this approach is that only pairwise matching is required between feature points instead of long consistent tracks. However, the piecewise planarity constraint can be restrictive and temporal smoothness is only imposed via post-processing which can lead to flickering.

Later, Taylor *et al*. [107] proposed a piecewise approach that uses locally rigid motion, reducing the number of points per patch to the minimum possible of three. Delaunay triangulation is applied to the image features to divide them into a set of triangles, which are reconstructed independently, using a linear algorithm, to form a *triangle soup*. Triangles that do not behave

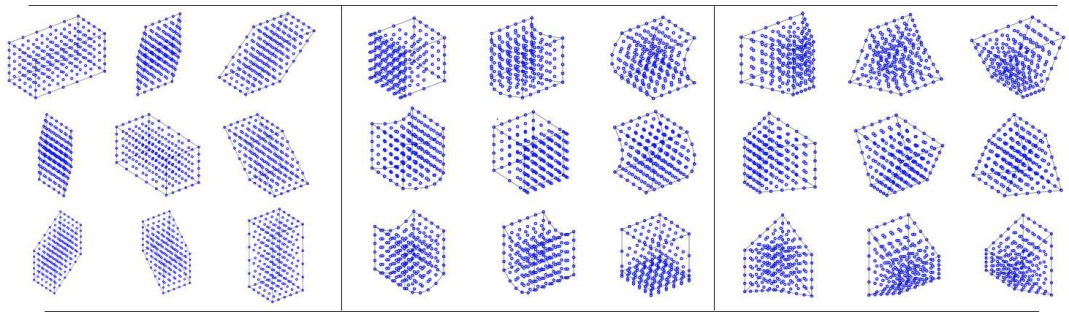


Figure 2.14: Some deformations encoded by the quadratic model. A cube is used as rest shape, the effects of linear, quadratic, and cross-terms are shown independently. This model generates deformations such as stretching, bending, sheering and twisting. Image courtesy of João Fayad.

rigidly are rejected using a predefined threshold on the reprojection errors, which allows the method to deal with outliers. In order to align the reconstructed triangles to provide a smooth 3D surface, a disambiguation step is then needed to solve for the relative depths and reflection ambiguities that results in an NP-hard problem to which an approximate solution is proposed. This grouping step makes the method applicable to non-rigid bodies that lose connectivity (for instance a paper tearing in two pieces).

In recent work Fayad *et al*. [41] proposed to use a more descriptive 3D model for the local patches also within a piecewise framework. The quadratic deformation model [42] was shown to be a better local model to reconstruct the individual patches than Varol *et al*.'s local planar model. It encapsulates three modes of deformation: *linear* which accounts for sheer and stretch; *quadratic* for bending and *mixed* terms for twist. In Figure 2.14 we show a visualisation of the effect of applying each deformation mode separately to a cube-shaped object. The quadratic model for each patch is optimised individually over the entire sequence using temporal smoothness priors. The patches are then aligned to give a global smooth surface using the overlapping points to impose global 3D consistency. In essence, this method is similar to Varol *et al*.'s [120] with the difference that the increased complexity of the quadratic model and its inherent temporal smoothness allows the smooth modelling of stronger deformations.

Fayad *et al*.'s method has recently been improved to drop the requirement of manual division of

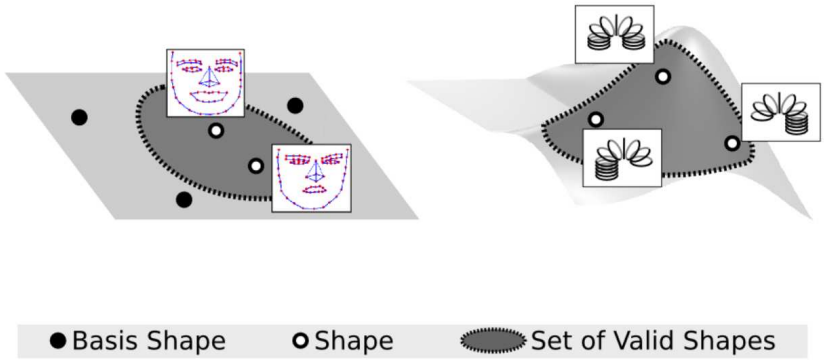


Figure 2.15: Rethinking non-rigid structure from motion [94]: the linear subspace shape model (left) cannot describe all deformations. Modelling non-rigid deformations as a piecewise linear manifold allows for more complex deformations to be explained. Image courtesy of Vincent Rabaud and Serge Belongie

the surface into patches. In [95] Russell *et al*. reformulate the NRSfM problem as a labelling one which allows to optimise jointly both the assignment of points to local models and the fitting of models to points to minimise a geometric cost (image reprojection error) using a variant of the graph-cut based algorithm α-expansion. This method is shown to obtain state of the art results outperforming all previous piecewise reconstruction methods. The same inference engine has also been extended to deal with the joint segmentation and 3D reconstruction of articulated objects in [43].

2.10.2 Manifold Learning

The work by Rabaud and Belongie [94] also departs from the classical low-rank factorisation with basis shapes. The main question posed concerning the low-rank basis shape model, was whether or not a linear manifold can represent accurately non-rigid shape. The authors argue that this is not the case for objects that undergo strong deformations, and propose to model the non-rigid deformations manifold as piecewise linear assuming that only small neighbourhoods of shapes are well modelled by a linear subspace. Figure 2.15 depicts the difference between

the linear subspace shape model and the proposed piecewise linear non-rigid shape manifold, showing that the latter can represent more complex non-rigid motions. Their approach relies on the concept of repetition: given a long sequence of a non-rigid shape, similar rigid shapes will appear in various instances along the sequence but seen from different viewpoints. Images are grouped into clusters that represent the same rigid shape and a manifold learning technique is then applied to learn the non-rigid shape manifold constraining the degrees of freedom of the object. Although they claim that they can deal with complex non-linear deformations, their test sequences are not as challenging as those attempted by the piecewise reconstruction methods [107, 41, 95].

Zhu *et al*. [136] propose a related approach in which the set of images is also grouped into clusters that represent the same rigid shape up to a rigid transformation. In practice, the epipolar constraint, or the trifocal tensor, can be used to estimate if images were generated by the same rigid object and belong to the same cluster. Zhu *et al*. introduce the concept of a *model graph* which greatly reduces the computational cost of discovering groups of images that represent the same rigid shape. The 3D shape for each image is then built by traversing this graph using their model-evolution algorithm based on incremental rigid SFM. Finally, a compressive sensing representation is used to model large deformations. Using a sparse basis, estimated by reducing the number of models in the shape clusters, allows to encode large non-linear deformations. One of the interesting contributions of this work is its application not just to sequences of non-rigid motion but also to large collections of photographs of similar but not identical objects in a category such as different types of cars.

2.10.3 Reconstruction in Trajectory Space

The low-rank shape basis model of Bregler *et al*. [15] explores the spatial properties of non-rigid motion, introducing rank constraints on the 3D location of the set of points (shape) at any given frame. The dual formulation of this model, proposed by Akhter *et al*. [6, 7], states that the rank constraint can be instead applied to the 3D trajectories of each individual point, modelling

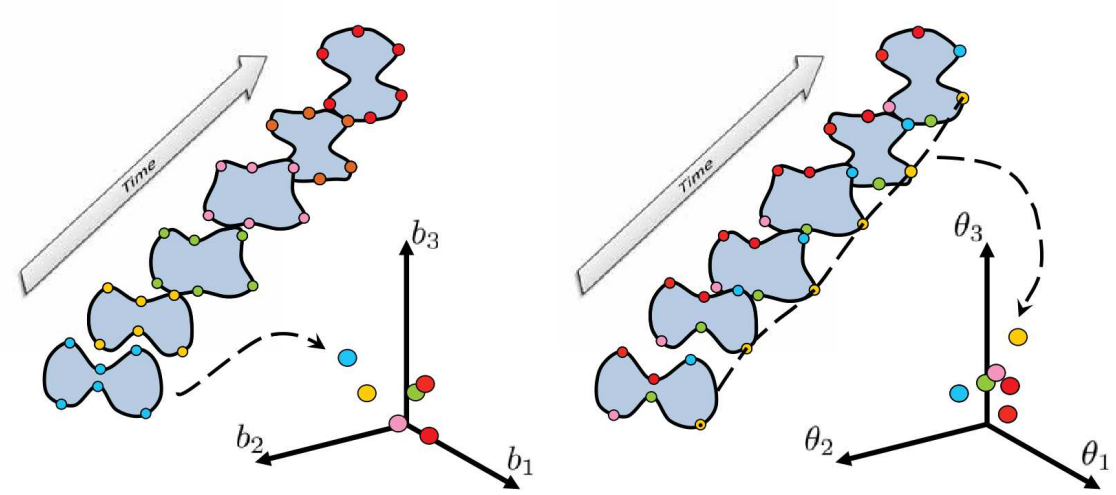


Figure 2.16: Dual representations for non-rigid objects. Left: a point configuration in space is a point in the low-dimensional space defined by basis shapes. Right: the same shape represented in trajectory space, each point trajectory in time is a point in a low-dimensional space defined by trajectory basis. Figure courtesy of Sohaib Khan and Yaser Sheikh

them as a linear combination of *basis trajectories*. Under the low-rank shape basis model, the basis shapes must be estimated for every new sequence. The advantage of the trajectory basis formulation is the ability to use a pre-defined trajectory basis for any sequence. This simplifies the problem greatly, leaving only the camera matrices and trajectory coefficients to be estimated. In their work, Akhter *et al*. chose to model the trajectories using Discrete Cosine Transform (DCT). While this choice might seem to restrict the types of trajectories it can represent, in [7] they showed that the DCT basis approximates well the expressive power of the PCA basis. Since the DCT is a basis for continuous functions, this model makes the implicit assumption that the 3D trajectories on the surface of the object are smooth in time. The estimation of the camera matrices and trajectory coefficients fits well within the factorisation framework and the resulting upgrade matrix is estimated by ensuring that the camera matrices are orthonormal. The method has been tested primarily on human motion capture data (for instance the CMU mocap data-set). In these cases (which in fact represent mostly articulated data), this method outperforms algorithms based on the dual shape-basis model such as Torresani *et al*.'s [111], but instead

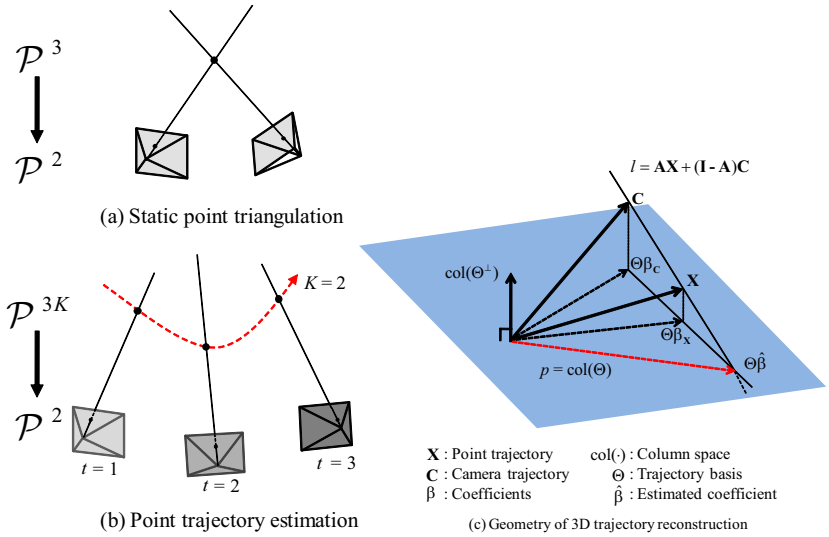


Figure 2.17: Reconstructibility result: a point trajectory of a non-rigid point (b) can only lie in the linear subspace defined by the DCT basis used (c). Park *et al.* [88] show that the recovered trajectory lies on the line connecting camera and point trajectory. This implies that the best result for the reconstruction $\Theta\beta_X$ is obtained when the camera trajectory is not correctly modelled by the DCT subspace. (Figure courtesy of Hun-Soo Park)

produces less accurate results on pure non-rigid data, such as deformations on human faces.

Park *et al.* [88], also reconstruct non-rigid motion using the DCT trajectory basis, but reconstructing each point trajectory independently, assuming a calibrated camera. The most interesting contribution of their work is the theoretical result that reconstructing a single trajectory works best when the camera motion does not lie on the subspace defined by the trajectory basis used to represent the 3D point trajectory. This *reconstructibility theorem* provides insight into reconstruction in trajectory space. Figure 2.17 depicts the reconstructibility theorem. Park *et al.* [88] show that the recovered trajectory lies on the line connecting the camera and point trajectories. This implies that the best result for the reconstruction of a 3D point trajectory (which must lie on the subspace by construction) is obtained when the camera trajectory does not lie on the same subspace, i.e. it is not correctly modelled by the DCT basis. Hence low-rank, low-frequency (smooth) 3D point trajectories that are well represented by the DCT basis will be only recovered accurately if captured by a camera moving randomly (high frequency cam-

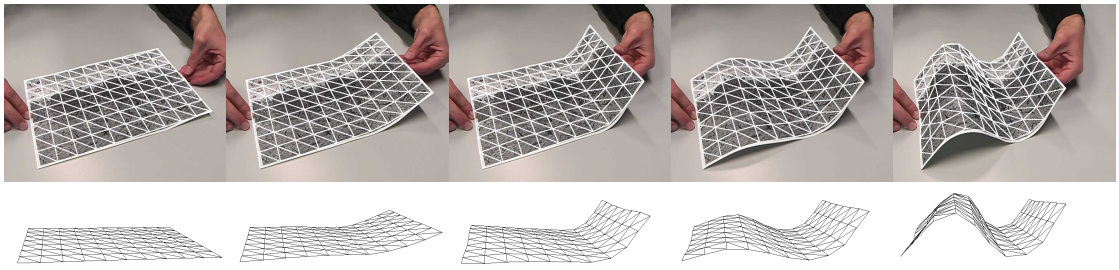


Figure 2.18: Results from the template-based method in [97] for 3D recovery of deformable surfaces. The vertices of a 3D triangle mesh are known in a reference frame, the 3D vertices are estimated from image 2D-3D correspondences. Each frame is treated separately. Images courtesy of Mathieu Salzmann

era trajectory). An experimental setup in which such assumption is true, is that of a crowd of photographers imaging an event from many different viewpoints.

Gotardo and Martinez [45] also make use of the DCT basis to model a smooth trajectory, but rather than modelling the trajectory of the 3D point in space, the basis is used to represent the smooth time-varying coefficients of a low-rank basis shape model. The matrices of coefficients and camera motion are expressed in a compact representation in the DCT domain. This allows the decoupling of the rank of the 3D shape basis and the rank of the DCT basis for the coefficients, making it possible to use high-frequencies for the coefficients, while keeping the number of basis shapes low. Their results show that, compared to Akhter *et al*.'s formulation [7], this approach can better model complex articulated deformation with higher frequency deformation components without the need to use a higher dimension subspace which could lead to over-fitting.

2.11 Template-based methods

A very successful alternative approach to monocular 3D reconstruction of deformable surfaces has developed in parallel to NRSfM. *Template-based* reconstruction of non-rigid surfaces assumes a given *reference* image in which the shape of the 3D surface is known in advance [97, 98, 90, 16]. The problem is then to infer the 3D shape in an *input* image in which the shape

has deformed. The method assumes that 2D correspondences exist between features in the reference and input images. Template-based reconstruction is therefore formulated for image-pairs. Naturally, it can be extended to process a long video by establishing correspondences independently between the reference and each input frame in the sequence. The implication is that it does not require long frame to frame correspondences in an image sequence as NRSfM methods do which allows for increased robustness. However, its disadvantages with respect to NRSfM are the increased difficulty in imposing temporal smoothness priors and the strong requirement of a known 3D shape template.

The surface is commonly represented as a triangulated mesh, as shown in Figure 2.18. The template is the 3D position of all vertices of the mesh in the reference image. Normally, mesh vertices are not found directly in the target image. Let $\mathbf{q}$ be a generic 3D point in the reference mesh, its coordinates can be expressed in terms of the nearest vertices as : $\mathbf{q} = [a\mathbf{v}_1, b\mathbf{v}_2, c\mathbf{v}_3]^T$, where $[a,b,c]^T$ are the *barycentric coordinates* of that point, $\mathbf{v}_i$, $1 \leq i \leq 3$ are the (known) 3D vertices of the mesh triangle where point $\mathbf{q}$ lies. Feature point matching between the target image and the reference image allows to compute the barycentric coordinates of such points. The reconstruction problem becomes the recovery of all vertices, given knowledge of 2D reprojection of the set of feature points (each feature point lying in one of the mesh triangles, with known barycentric coordinates). If the only constraint imposed on the surface are the point correspondences, it is possible to obtain a reconstruction with accurate 2D point reprojection, but incorrect 3D shape. Additional constraints are required to constrain the 3D coordinates of the surface.

It was first shown by Salzmann *et al*. [97] that in the context of template-based non-rigid 3D shape reconstruction it is possible to formulate the minimisation of image reprojection error as a convex SOCP optimisation problem. The additional constraint used to prevent an under-constrained solution is to enforce temporal consistency, disallowing large changes of edge orientation between consecutive frames which can be expressed as additional SOCP constraints, yielding a convex formulation. This convex approach still relied on a full video sequence and the availability of frame to frame correspondences.

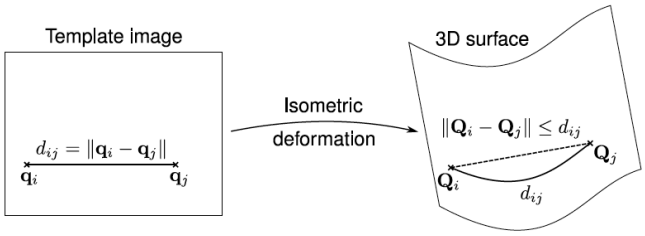


Figure 2.19: The Euclidean distance between point pairs will be smaller than the geodesic distance along the surface. Equality constraints can be relaxed with inequality constraints, this allows a convex formulation. Figure courtesy of Adrien Bartoli.

In later work the temporal consistency constraints were replaced by geometric ones that permitted to perform 3D reconstruction using a single input image. In practice, the constraints used describe assumptions on the allowable surface deformations. It was shown in [98] that recovering the 3D shape of a flexible *inextensible* surface from 3D-to-2D correspondences can be achieved in closed form by solving a set of quadratic equations by simply constraining the distances between selected surface points to remain constant. This method was restricted to smooth surfaces. Later, a new convex formulation was proposed to deal with sharply folding surfaces [96] by replacing the distance equality constraints between surface points with inequality ones that are convex and allow points to come closer to each other but prevent them from moving further apart than their geodesic distance. This constraint can be visualised in Figure 2.19. Inextensibility constraints have been further exploited by other authors who have represented the smooth surface using thin-plate splines [90], or free-form deformations [16] and exploited the orthonormality condition that the 2D-3D isometry map induces on the Jacobian matrix [16].

While the field of template-based reconstruction is now quite mature and well understood and robust methods exist for monocular 3D reconstruction of deformable surfaces, the strong assumption of a known 3D shape makes NRSfM an attractive alternative when no prior information is available about the surface or the way it deforms.

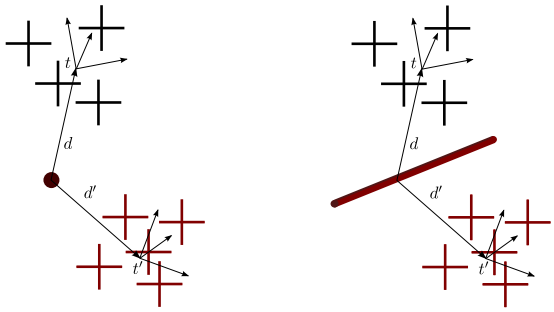


Figure 2.20: Articulated motion: 3D points belonging to different objects are forced to move around a common joint centre (left), or rotate around a common hinge (right). Each object is described in its own reference frame, the distances $\mathbf{d}$ and $\mathbf{d}'$ between each centre of mass and the joint centre must remain constant. In addition, each centre of mass will show a global translation $\mathbf{t}$,$\mathbf{t}'$ on the image plane.

2.12 Articulated Structure from Motion: A-SfM

Articulated motion has also been recently formulated using a structure from motion approach [113, 131]. The key idea is to model the articulated motion space as a set of intersecting motion subspaces — the intersection of two motion subspaces implies the existence of a link between the parts. Articulation constraints can then be imposed during factorisation to recover the location of joints and axes on the image plane. Tresadern and Reid [113] propose a metric upgrade providing joint information in 3D, approximating the upgrade as a linear problem, thus obtaining a closed-form solution. One of the most important assumptions of these methods is that segmentation of the tracking data into the different articulated parts must be known in advance. This problem, known as motion segmentation, has received substantial interest (see for example Vidal and Hartley [122]). The first solution to this problem was proposed by Costeira and Kanade [28], for multiple independent rigid motions. Recently, Yan and Pollefeys [132] have proposed a more general solution able to provide segmentation for objects undergoing articulated motion.

2.12.1 Articulated Shape Model

In the case of articulated structure, the relative motions of the segments that form an articulated body are dependent and this results in a drop in the dimensionality of the measurement matrix

$\mathtt{W} = \left[\begin{array}{c|c} \mathtt{W}_1 & \mathtt{W}_2 \end{array}\right]$ that contains the 2D image points of the two segments. In the case of a *universal joint* the two shapes share a common translation (i.e. the distance between shapes and joint is constant) while in the case of a *hinge joint* the shapes also share a common rotation axis. Both the work of Yan and Pollefeys [131] and that of Tresadern and Reid [113] provide a solution to the recovery of articulated motion, and have been developed independently. We will briefly describe both methods.

2.12.2 Subspace analysis

Yan and Pollefeys [131, 133] proposed a method to analyse articulated motion and recover joint and axis positions. Consider two independently moving rigid objects imaged by a single camera. Let $\mathtt{W}_1$ and $\mathtt{W}_2$ be the measurement matrices containing 2D feature tracks for the two objects respectively. Under an affine camera model, the combined measurement matrix $\mathtt{W} = [\mathtt{W}_1|\mathtt{W}_2]$ can be written as:

$$\mathtt{W} = [\mathtt{W}_1|\mathtt{W}_2] = [\mathtt{M}_1|\mathbf{t}_1|\mathtt{M}_2|\mathbf{t}_2] \begin{bmatrix} \mathtt{S}_1 & 0 \\ 0 & \mathtt{S}_2 \end{bmatrix} \tag{2.30}$$

Where each object is associated with a motion matrix $[\mathtt{M}|\mathbf{t}]$, containing the $2F \times 3$ affine camera matrices for all frames (F being the number of frames), and the $2F \times 1$ translation vector $\mathbf{t}$. In order to incorporate the translation, the shape matrices are augmented with a row vector of ones, each shape matrix will thus have 4 rows. Equation 2.30 implies that $\mathtt{W}$ is at most of rank 8.

Let us consider the case where the two objects $\mathtt{S}_1$ and $\mathtt{S}_2$ are coupled by a *universal joint*. The distance between each object centroid and the joint rotation centre is a constant (cfr Figure 2.20). Hence, without loss of generality, the world coordinate system can be chosen such that all points of the first object $\mathtt{S}_1$ are fixed in 3D, and with its origin on the location of the joint centre. In this coordinate system, the second object at a generic frame f can be expressed as:

$$\mathtt{S}_{2f} = \begin{pmatrix} \bar{\mathtt{R}}_f & 0 \\ 0 & \mathbf{1} \end{pmatrix} \mathtt{S}_2$$

That is, the second object can only rotate around the joint centre, with $\bar{\mathtt{R}}_f$ being the relative rotation between the two objects at frame f. In this coordinate system, the two objects also share the same translation vector, since the distance between each object and the joint centre is a constant. For each frame the measurements can be written as:

$$[\mathtt{W}_1|\mathtt{W}_2]_f = [\mathtt{M}_{1f}|\mathbf{t}_{1f}|\mathtt{M}_{2f}|\mathbf{t}_{1f}] \begin{bmatrix} \mathtt{S}_1 & 0 \\ 0 & \mathtt{S}_{2f} \end{bmatrix} = [\mathtt{M}_{1f}|\mathbf{t}_{1f}|\mathtt{M}_{1f}\bar{\mathtt{R}}_f|\mathbf{t}_{1f}] \begin{bmatrix} \mathtt{S}_1 & 0 \\ 0 & \mathtt{S}_2 \end{bmatrix} \tag{2.31}$$

Stacking equation 2.31 for all frames results in the motion matrix containing two copies of the $2F \times 1$ translation $\mathbf{t}$. This common column implies that the motion matrix is rank deficient, having at most rank 7 for the case of a universal joint.

In the case of a *hinge joint*, the z axis of the world coordinate system can be aligned with the hinge, resulting in a relative rotation $\bar{\mathtt{R}}_f$ of the form:

$$\bar{\mathtt{R}}_f = \begin{bmatrix} \cos\theta_f & \sin\theta_f & 0 \\ -\sin\theta_f & \cos\theta_f & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.32}$$

where θ_f is the angle between the two objects at frame f. This implies a further drop in dimensionality: Equation 2.31 implies a motion matrix with two duplicated columns, the translation vector $\mathbf{t}$ and the last column of the matrices $\mathtt{M}_1$ and $\mathtt{M}_2$. This implies a motion matrix of at most rank 6.

The drop in rank of the motion matrix when two objects are joined by an articulation means that the subspaces spanned by the columns of the measurement matrices intersect. The intersection of the subspaces can be one or two-dimensional, respectively for the case of a universal or hinge joint. Yan and Pollefeys [131] have shown that the intersection of the subspaces is the motion subspace of the joint. They show that this property can be used to recover the 2D trajectory of the joint centre for the case of universal joint, and the 2D trajectories of two points on the axis, in the case of a hinge joint. In work done in parallel, the same result on the dimensionality of

the motion subspace was obtained by Tresadern and Reid [113], who also proposed a metric upgrade step, to recover joint trajectories in 3D.

2.12.3 Joint estimation in 3D

Tresadern and Reid's factorisation approach [113] can recover articulated structures in 3D, together with 3D position of joints and axes. The spatial relationship between the barycentre of each (rigid) object and the joint centre can be written as:

$$\mathbf{t}^{(1)} + \mathtt{R}^{(1)}\mathbf{d}^{(1)} = \mathbf{t}^{(2)} + \mathtt{R}^{(2)}\mathbf{d}^{(2)} \tag{2.33}$$

where $\mathbf{t}^{(1)}$ and $\mathbf{t}^{(2)}$ are the 2D image centroids of the two objects, $\mathtt{R}^{(1)}$ and $\mathtt{R}^{(2)}$ the 2×3 orthographic camera matrices and $\mathbf{d}^{(1)}$ and $\mathbf{d}^{(2)}$ the 3D displacement vectors of each articulation link from the joint centre. The constraint expressed in equation (2.33) results in a reduced dimensionality of the motion and shape subspaces. The geometric relationship expressed by this equation can be visualised in Figure 2.20. The 3D recovery is formulated as a factorisation problem. In the case of a universal joint, the distances between object centroid and joint centre $\mathbf{d}^{(1)}$,$\mathbf{d}^{(2)}$ are constant. The existence of a link implies that objects translate in space together — if the vectors $\mathbf{d}^{(1)}$ and $\mathbf{d}^{(2)}$ were known, the measurement matrix $[\mathtt{W}_1|\mathtt{W}_2]$ could be written as:

$$[\mathtt{W}_1|\mathtt{W}_2] = \mathtt{MS} = \begin{bmatrix} \mathtt{M}^{(1)} & \mathtt{M}^{(2)} & \mathbf{t}^{(1)} \end{bmatrix} \begin{bmatrix} \mathtt{S}^{(1)} & \mathbf{d}^{(1)} \\ 0 & \mathtt{S}^{(2)} - \mathbf{d}^{(2)} \\ \mathbf{1}^T & \mathbf{1}^T \end{bmatrix} \tag{2.34}$$

where $\mathtt{S}$ is a rank-7 matrix, containing three rows for the coordinates of each object, and a row of ones for the common translation vector. Equation 2.34 expresses in matrix form the relationship of equation 2.33 in all frames. In the case of a universal joint, Tresadern and Reid [113] propose a factorisation method to solve for the shape matrices and the unknown length of vectors $\mathbf{d}^{(1)}$ and $\mathbf{d}^{(2)}$.

In the case of a hinge joint, without loss of generality, the rotation axis can be made to coincide with the x axis of the world coordinate system, and any point along the rotation axis can be picked as the joint centre (cfr Figure 2.20). The measurement matrix can again be factorised into the product of a motion and a shape matrix, where the shape matrix $\mathtt{S}$ encapsulates the 3D coordinates for both objects, arranged in such a way as to enforce a common axis:

$$\mathtt{S} = \begin{bmatrix} x_1^{(1)} & \cdots & x_{P_1}^{(1)} & x_1^{(2)} & \cdots & x_{P_2}^{(2)} \\ y_1^{(1)} & \cdots & y_{P_1}^{(1)} & 0 & \cdots & 0 \\ z_1^{(1)} & \cdots & z_{P_1}^{(1)} & 0 & \cdots & 0 \\ 0 & \cdots & 0 & y_1^{(2)} & \cdots & y_{P_2}^{(2)} \\ 0 & \cdots & 0 & z_1^{(2)} & \cdots & z_{P_2}^{(2)} \end{bmatrix} \tag{2.35}$$

where the first articulated link has P_1 3D points and the second one has P_2. It is clear that the rank of the measurement matrix must be constrained to be at most 6:

$$[\mathtt{W}_1|\mathtt{W}_2] = [\mathtt{M}_1|\bar{\mathtt{M}}_2|\mathbf{t}_1] \begin{bmatrix} \mathtt{S}_{1x} & \mathtt{S}_{2x} + \mathbf{d}_x^{(1)} + \mathbf{d}_x^{(2)} \\ \mathtt{S}_{1y} & \mathbf{d}_y^{(1)} \\ \mathtt{S}_{1z} & \mathbf{d}_z^{(1)} \\ 0 & \mathtt{S}_{2y} + \mathbf{d}_y^{(2)} \\ 0 & \mathtt{S}_{2z} + \mathbf{d}_z^{(2)} \\ \mathbf{1}^T & \mathbf{1}^T \end{bmatrix} \tag{2.36}$$

The zero blocks in the shape matrix are substituted with the block-replicated coordinates of the vectors $\mathbf{d}^{(1)}$ and $\mathbf{d}^{(2)}$, which express distances between the joint centre and the object centroid in 3D. This matrix formulation express the joint constraint of equation 2.33, with the added constraint that the two motion matrices have a common column: in this case, the common column is relative to the x axis, meaning that $\bar{\mathtt{M}}_2$ is a $2F \times 2$ block containing only two columns of the motion matrix for the second object, the ones relative to the y and z axis.

To solve for 3D structure and motion with a factorisation approach the method of Tresadern and

Reid [113] performs a rank 5 SVD decomposition on the registered measurement matrix, then enforces the structure of the shape matrix by premultiplying the SVD factors by an invertible matrix composed of the null space of the shape SVD factor. The recovered affine reconstruction needs to be upgraded to metric by enforcing the metric constraints. Although the constraints are non-linear Tresadern and Reid propose a linear approximation for recovering the hinge joint in 3D.

In chapter 3 we propose a new framework to impose the exact non-linear metric constraints. Taking the linear estimate as initialisation, we propose a factorisation algorithm that projects the solution onto the correct manifold of constraints.

2.13 Closure

This chapter discussed the literature in the field of 3D shape estimation from monocular sequences focusing on factorisation approaches to non-rigid and articulated structure from motion. In NRSfM, we have shown that the problem is inherently under-constrained and intractable without the use of additional priors or constraints. We have shown how most of the efforts in the NRSfM community have gone into solving the inherent metric ambiguity. Imposing the metric constraints results in a non-linear estimation problem which requires a good initialisation and the use of priors to avoid local minima. The problem becomes even harder in the presence of noise or missing data due to occlusions. We have provided a taxonomy of methods for NRSfM where we divide approaches according to the shape model used and to the optimisation technique employed to estimate the parameters. Additionally we have described the different types of *statistical* and *physical* priors used to avoid ambiguous solutions. We have also described current approaches to articulated SfM and noted that only a linear approximation of the metric upgrade is estimated.

Although different methods have been proposed in the literature to tackle the metric upgrade, it still remains an open problem. While closed-form approaches find the correct solution in the noise free case, they are extremely sensitive to noise, perform poorly in real sequences

and cannot deal with missing data. On the other hand, alternation and non-linear optimisation techniques impose the metric constraints via parametrisation and require strong priors. Also, often imposing the constraints requires approximations.

The first part of this thesis focuses on new strategies to impose the metric constraints. First we show a unified approach to non-rigid and articulated factorisation. We propose a common alternating bilinear approach to solve for 3D shape and motion, associated with a projection step onto the manifolds of (respectively) non-rigid and articulated metric constraints to ensure that the solutions satisfy the metric constraints. Then we present a bilinear factorisation approach that completely decouples the bilinear estimation step from the projection onto the manifold of acceptable solutions. We show extensive experimental evaluations on ground truth and real sequences which show that we are able to deal with high percentages of missing data.

The final part of our work aims at pushing non-rigid structure from motion solutions towards the sequential domain, a scenario in which reconstruction can be obtained during image acquisition. Currently, all NRSfM methods are batch: all the frames are processed at once after the acquisition takes place. In the final chapter we describe our incremental approach to the estimation of deformable models. Image frames are processed on-line in a sequential fashion. The shape-model is also built on-line with new modes added incrementally when the current model cannot model a new image well enough.

Chapter 3

Metric Projections for Deformable and Articulated Structure-From-Motion

---

Most approaches to deformable and articulated structure from motion require to upgrade an initial affine solution to Euclidean space by imposing metric constraints on the motion matrix. While in the case of rigid structure the metric upgrade step is simple since the constraints can be formulated as linear, deformability in the shape introduces non-linearities. We propose an alternating bilinear approach to solve for non-rigid 3D shape and motion, associated with a globally optimal projection step of the motion matrices onto the manifold of metric constraints. We will present an algorithm for recovering the 3D shape and motion of deformable and articulated objects purely from uncalibrated 2D image measurements using a factorisation approach. Our novel optimal projection step combines into a single optimisation the computation of the orthographic projection matrix and the configuration weights. We avoid the difficult problem of metric upgrade by projecting the solution to the *motion manifold*. We define the *motion manifold* as the set of matrices that satisfy the metric constraints. The projection gives the closest motion matrix that satisfies the correct block structure with the additional constraint that the projection

matrix have orthonormal rows (*i.e.* its transpose lies on the Stiefel manifold). This constraint turns out to be non-convex. The key contribution of this work is the solution to the non-convex projection step. We present a tight convex relaxation that obtains the global optimum, and then introduce an efficient convex relaxation which speeds up the computation while preserving accuracy. Efficient in the sense that, for both the cases of deformable and articulated motion, the proposed relaxations turned out to be exact (*i.e.* tight) in all our numerical experiments. The convex relaxations are semi-definite (SDP) or second-order cone (SOCP) programs which can be readily tackled by popular solvers. An important advantage of these new algorithms is their ability to handle missing data which becomes crucial when dealing with real video sequences with self-occlusions. We show successful results of our algorithms on synthetic and real sequences of both deformable and articulated data. We also show comparative results with state of the art algorithms which reveal that our new methods outperform existing ones.

3.1 Introduction

The combined inference of the motion of a camera and the 3D geometry of an unconstrained scene viewed solely from a sequence of images is a long-standing challenge for the Computer Vision community. The fundamental assumption which has allowed robust solutions to the problem is that of scene rigidity. However, when dealing with image objects that vary their 3D shape, the Structure From Motion (SfM) problem becomes inherently ambiguous and non-linear. The seminal work of Bregler *et al*. [15] was the first to deal with the case of deformable objects viewed by a single camera. Their key insight was to use a low-rank shape model to represent the deforming shape as a linear combination of k basis shapes which encode its main modes of deformation. This model not only provided an elegant extension of the rigid factorisation framework devised by Tomasi and Kanade [110] but has also opened up new computational and theoretical challenges in the field.

Although this low-rank shape model has proved a successful representation, the Non-Rigid Structure from Motion (NRSfM) problem is inherently under-constrained. Most approaches

formulate it as an optimisation problem where the objective function to minimise is the image reprojection error. Recent methods focus on overcoming the problems caused by ambiguities and degeneracies by proposing different optimisation schemes and the use of generic priors. In the previous chapter we discussed how most of the efforts in the NRSfM community have gone to solve the inherent metric ambiguity. Imposing the orthonormality constraint results in a non-linear estimation problem which requires a good initialisation and the use of priors to avoid local minima. The problem becomes even harder in the presence of noise or missing data due to occlusions.

Articulated motion has also been recently formulated using a structure from motion approach [113, 133] modelling the articulated motion space as a set of intersecting motion subspaces — the intersection of two motion subspaces implies the existence of a link between the parts. Articulation constraints can then be imposed during factorisation to recover the location of joints and axes. While Yan and Pollefeys only compute the location of joints and axes on the image plane and do not perform a 3D reconstruction, Tresadern and Reid go further and compute the metric upgrade, but only recover a linear approximation of the correcting transformation [113]. Both approaches require full data and therefore cannot deal with missing tracks, a situation that commonly occurs for instance when tracking humans.

3.1.1 Contributions

In this chapter we present a new unified approach to perform the metric upgrade in the cases of articulated and deformable structure viewed by an orthographic camera in the presence of missing data.

In the non-rigid case our approach is most closely related to the trilinear schemes of Torresani *et al*. [112] and Wang *et al*. [123] we described in Chapter 2. Both approaches use an identical alternating least squares framework to estimate the configuration weights, basis shapes and orthographic camera matrices, solving iteratively for each of the unknowns leaving the others fixed. The only difference between these two approaches is in the way that the orthographic

camera matrices are updated and the metric constraints imposed – the other two steps in the alternation minimise the same cost.

While Torresani *et al*. enforce the exact metric constraints through an exponential map parametrisation of the rotation matrices, the update of the camera matrix is only an approximation — the camera matrix cannot be updated in closed form and instead they perform a single Gauss-Newton step. Alternatively, in their Rotation Constrained Powerfactorization algorithm (RCPF) Wang *et al*.first update the orthographic camera matrix via least squares and an additional step is incorporated to project it onto the Stiefel manifold via its SVD decomposition. This simple projector is in fact identical to the one proposed by [70] for the case of rigid structure. Finally, in order to deal with missing data the above trilinear approaches [112, 123] resort to using only the available image tracks in their alternating scheme.

Similarly to Torresani *et al*. and Wang *et al*. we also propose an iterative alternating scheme to solve the non-rigid structure from motion problem. However, our optimisation scheme is bilinear, alternating between the estimation of the motion and the shape matrices, with an additional projection step of the motion matrices onto the manifold of metric constraints. At the expense of solving a more complex optimisation problem, our efficient convex relaxation provides an optimal minimiser to solve simultaneously for the orthographic camera matrix and configuration weights that give a motion matrix that satisfies the appropriate block structure while also ensuring that the orthographic camera matrix satisfies the constraint of having orthonormal rows (its transpose lies on the Stiefel manifold[1]). Here and throughout the chapter, the optimal projection of a matrix onto a given set of matrices, denotes the closest point on that set from the given matrix with respect to the Frobenius norm. Extensive tests carried out on motion capture sequences with ground truth 3D data, reported in Section 3.5, show that adding a projection step (Wang *et al*.'s or ours) improves greatly the results obtained in the case of missing data with respect to other methods. However, even better improvements are achieved when using our

[1]The Stiefel manifold $\mathtt{V}_{k,m}$ may be viewed as the collection of all $m \times k$ matrices whose columns form an orthonormal set. More precisely, the (real) Stiefel manifold $\mathtt{V}_{k,m}$ is the collection of all ordered sets of k orthonormal vectors in Euclidean space $\mathbb{R}^m$.

bilinear algorithm associated with the proposed metric projection instead of Wang *et al.*'s [123] trilinear scheme and simpler projector.

In order to deal with missing data, our algorithm performs an outer iterative loop in which, at each step of the iteration, we run our non-rigid factorisation algorithm and we use the new estimates of the rotations, translations, basis shapes and coefficients to provide a new estimate of the missing data. Our experimental tests shown in Section 3.5 reveal that dealing with incomplete tracks using this outer loop allows to cope with much higher percentages of missing data than the trilinear approaches [112, 123] that only use the available data.

In summary, we see three substantial contributions in our approach. First, in contrast to their trilinear schemes, our optimisation scheme is bilinear, alternating between the estimation of the motion and the shape matrices. Secondly, our novel optimal projection step combines into a single optimisation the computation of the camera matrix and the configuration weights that give the closest motion matrix that lies on the non-rigid *motion manifold* with the additional constraint that the camera matrix is guaranteed to have orthonormal rows (*i.e.* its transpose lies on the Stiefel manifold). Finally, our experiments reveal that dealing with missing data using an iterative outer loop to re-estimate the missing entries greatly improves the results with missing data.

The notion of *motion manifolds* has been recently introduced in the case of rigid shapes by Marques and Costeira [70]. Our work extends and generalises it to the case of deformable and articulated shapes. In particular, we impose that the camera matrix has orthonormal rows, therefore its transpose lies on the $\mathtt{V}_{2,3}$ Stiefel manifold.[2]. This constraint results in a non-convex problem which we were able to solve by a convex relaxation in the case of deformable shape. In the articulated case, we efficiently compute the joints given the non-linear constraints on the motion of the two bodies. The result is an algorithm where the recovered motion matrices have the exact orthogonality constraints imposed. One of the main advantages of our approach is that

[2]The Stiefel manifold $\mathtt{V}_{k,m}$ may be viewed as the collection of all $m \times k$ matrices whose columns form an orthonormal set. More precisely, the (real) Stiefel manifold $\mathtt{V}_{k,m}$ is the collection of all ordered sets of k orthonormal vectors in Euclidean space $\mathbb{R}^m$.

it can be extended naturally to deal with missing data in a similar way to [70].

As a final observation we should stress that, while most NRSfM algorithms proposed to date need to rely on the use of priors to solve for the 3D shape and the camera motion [8, 111] avoiding ambiguities, our new algorithms can obtain reliable solutions without having to impose priors such as smoothness on the camera motion or the deformations.

The contributions of this work have been published in [85, 86].

3.2 Factorisation for Structure from Motion

Consider the set of 2D image trajectories obtained when the points lying on the surface of a 3D object are viewed by a moving camera. Defining the non-homogeneous coordinates of a point j in frame i as the vector $\mathbf{w}_{ij} = (u_{ij}\ v_{ij})^\top$ we may write the measurement matrix $\mathtt{W}$ that gathers the coordinates of all the points in all the views as:

$$\mathtt{W} = \begin{bmatrix} \mathbf{w}_{11} & \dots & \mathbf{w}_{1p} \\ \vdots & \ddots & \vdots \\ \mathbf{w}_{f1} & \dots & \mathbf{w}_{fp} \end{bmatrix} = \begin{bmatrix} \mathtt{W}_1 \\ \vdots \\ \mathtt{W}_f \end{bmatrix} \tag{3.1}$$

where f is the number of frames and p the number of points.

The measurement matrix can be factorised into the product of two low-rank matrices as $\mathtt{W} = \mathtt{M}_{2f\times r}\ \mathtt{S}_{r\times p}$, where $\mathtt{M}$ and $\mathtt{S}$ correspond to the motion and shape subspaces respectively. As a result, the rank of $\mathtt{W}$ is constrained to be $\text{rank}\{\mathtt{W}\} \leq r$ where $r \ll \min\{2f, p\}$. The rank of these subspaces is dictated by the properties of the camera projection and the nature of the shape of the object being observed (rigid, deformable, articulated, etc.). This rank constraint forms the basis of the factorisation method for the estimation of 3D structure and motion.

Matrices $\mathtt{M}$ and $\mathtt{S}$ can be expressed as $\mathtt{M} = \left[\mathtt{M}_1^\top \cdots \mathtt{M}_f^\top\right]^\top$ and $\mathtt{S} = [\mathbf{S}_1 \cdots \mathbf{S}_p]$ where $\mathtt{M}_i$ is the $2 \times r$ camera matrix that projects the 3D shape onto the image frame i and $\mathbf{S}_j$ encodes the 3D coordinates of point j.

3.2.1 Rigid Shape

In the case of a rigid object viewed by an orthographic camera, if we assume the measurements in $\mathtt{W}$ are registered to the image centroid, the camera motion matrices $\mathtt{M}_i$ and the 3D points $\mathbf{S}_j$ can be expressed as: $\mathtt{M}_i = \begin{bmatrix} r_{i1} & r_{i2} & r_{i3} \\ r_{i4} & r_{i5} & r_{i6} \end{bmatrix} = \mathtt{R}_i$ and $\mathbf{S}_j = \begin{bmatrix} X_j & Y_j & Z_j \end{bmatrix}^\top$ where $\mathtt{R}_i$ is a 2×3 matrix whose transpose lies on the Stiefel manifold (i.e. a 3×2 Stiefel matrix), since $\mathtt{R}_i$ contains the first two rows of a rotation matrix (i.e. $\mathtt{R}_i\mathtt{R}_i^\top = \mathtt{I}_{2\times 2}$) and $\mathbf{S}_j$ is a 3-vector containing the metric coordinates of the 3D point. Therefore the rank of the measurement matrix is $r \leq 3$. The rigid *motion manifold* corresponds to the manifold of matrices with pairwise orthogonal rows.

3.2.2 Deformable Shape Model

In the case of deformable objects the observed 3D points change as a function of time. In this work we use the low-rank shape model defined in Bregler *et al*. [15] in which the 3D points deform as a linear combination of a fixed set of k rigid shape bases according to time varying coefficients. In this way, $\mathtt{S}_i = \sum_{d=1}^{k} l_{id}\mathtt{B}_d$ where the matrix $\mathtt{S}_i = [\mathbf{S}_{i1}, \cdots \mathbf{S}_{ip}]$ is the 3D shape of the object at frame i, the $3 \times p$ matrices $\mathtt{B}_d$ are the shape bases and l_{id} are the coefficients (sometimes called deformation weights). If we assume an orthographic projection model the coordinates of the 2D image points observed at each frame i are then given by:

$$\mathtt{W}_i = \mathtt{R}_i \left(\sum_{d=1}^{k} l_{id}\mathtt{B}_d \right) + \mathtt{T}_i \tag{3.2}$$

where the matrix $\mathtt{R}_i$ is 2×3 with orthonormal rows, such that $\mathtt{R}_i^\top$ is a *Stiefel matrix* and the $2 \times p$ matrix $\mathtt{T}_i$ aligns the image coordinates to the image centroid. The aligning matrix $\mathtt{T}_i$ is such that $\mathtt{T}_i = \mathbf{t}_i\mathbf{1}_p^\top$ where the 2-vector $\mathbf{t}_i$ is the 2D image centroid and $\mathbf{1}_p$ a vector of ones. When the image coordinates are registered to the centroid of the object and we consider all the frames in

the sequence, we may write the measurement matrix as:

$$\mathtt{W} = \begin{bmatrix} l_{11}\mathtt{R}_1 & \dots & l_{1k}\mathtt{R}_1 \\ \vdots & \ddots & \vdots \\ l_{f1}\mathtt{R}_f & \dots & l_{fk}\mathtt{R}_f \end{bmatrix} \begin{bmatrix} \mathtt{B}_1 \\ \vdots \\ \mathtt{B}_k \end{bmatrix} = \begin{bmatrix} \mathtt{M}_1 \\ \vdots \\ \mathtt{M}_f \end{bmatrix} \begin{bmatrix} \mathtt{B}_1 \\ \vdots \\ \mathtt{B}_k \end{bmatrix} = \mathtt{MS} \tag{3.3}$$

Since $\mathtt{M}$ is a $2f \times 3k$ matrix and $\mathtt{S}$ is a $3k \times p$ matrix in the case of deformable structure the rank of $\mathtt{W}$ is constrained to be at most $3k$. The motion matrices now have the form $\mathtt{M}_i = [\mathtt{M}_{i1} \dots \mathtt{M}_{ik}] = [l_{i1}\mathtt{R}_i \dots l_{ik}\mathtt{R}_i]$. Therefore, in the deformable *motion manifold* the motion matrices have a distinct repetitive structure and every 2×3 $\mathtt{M}_{ik}$ sub-block is composed of the transpose of a *Stiefel matrix* multiplied by a scalar.

3.2.3 Articulated Shape Model

In the case of articulated structure, the relative motions of the segments that form an articulated body are dependent and this results in a drop in the dimensionality of the measurement matrix $\mathtt{W} = \left[\begin{array}{c|c} \mathtt{W}^{(1)} & \mathtt{W}^{(2)} \end{array}\right]$ that contains the 2D image points of the two segments. In the case of a *universal joint* the two shapes share a common translation (i.e. the distance between the centres of mass of the shapes is constant) while in the case of a *hinge joint* the shapes also share a common rotation axis [113, 133]. Naturally, this approach requires that an initial segmentation stage has taken place to assign the trajectories in $\mathtt{W}$ to the respective shapes for which a solution was recently provided in [133].

In a *universal joint* [113] the distance between the centres of the two shapes is constrained to be constant (for instance, the head and the torso of a human body) but with independent rotation components. At each frame the shapes connected by a joint satisfy:

$$\mathbf{t}^{(1)} + \mathtt{R}^{(1)}\mathbf{d}^{(1)} = \mathbf{t}^{(2)} + \mathtt{R}^{(2)}\mathbf{d}^{(2)} \tag{3.4}$$

where $\mathbf{t}^{(1)}$ and $\mathbf{t}^{(2)}$ are the 2D image centroid of the two objects, $\mathtt{R}^{(1)}$ and $\mathtt{R}^{(2)}$ the 2×3 ortho-

graphic camera matrices and $\mathbf{d}^{(1)}$ and $\mathbf{d}^{(2)}$ the 3D displacement vectors of each shape from the joint. The relation in equation (3.4) gives the reduced dimensionality in the motion and shape subspaces. Thus, the shape matrix $\mathtt{S}$ can be written as:

$$\mathtt{S} = \begin{bmatrix} \mathtt{S}^{(1)} & \mathbf{d}^{(1)} \\ \mathtt{0} & \mathtt{S}^{(2)} - \mathbf{d}^{(2)} \\ \mathbf{1} & \mathbf{1} \end{bmatrix} \tag{3.5}$$

where $\mathtt{S}$ is a full rank-7 matrix. The motion for a frame i has to be accordingly arranged to satisfy equation (3.4) as:

$$\mathtt{M}_i = \begin{bmatrix} \mathtt{R}_i^{(1)} & \mathtt{R}_i^{(2)} & \mathbf{t}_i^{(1)} \end{bmatrix}. \tag{3.6}$$

In the case of a *hinge joint*, if we assume the image coordinates to be registered to the centroid of each segment, then the motion matrices $\mathtt{M}_i$ that lie on the articulated *motion manifold* can be written as:

$$\mathtt{M}_i = \begin{bmatrix} \mathbf{u}_i & \mathtt{A}_i & \mathtt{B}_i \end{bmatrix} \tag{3.7}$$

where $\mathbf{u}$ is the common rotation axis for both objects, $\mathtt{A}_i$ and $\mathtt{B}_i$ are 2×2 matrices such that $\begin{bmatrix} \mathbf{u}_i | \mathtt{A}_i \end{bmatrix}$ and $\begin{bmatrix} \mathbf{u}_i | \mathtt{B}_i \end{bmatrix}$ are the 2×3 camera matrices (with orthonormal rows) associated with the first and second shape respectively. The metric constraints in the case of a hinge can therefore be expressed as:

$$\begin{aligned} \begin{bmatrix} \mathbf{u}_i & | \mathtt{A}_i \end{bmatrix} \begin{bmatrix} \mathbf{u}_i^\top \\ \mathtt{A}_i^\top \end{bmatrix} &= \mathtt{I}_{2\times 2} \\ \begin{bmatrix} \mathbf{u}_i & | \mathtt{B}_i \end{bmatrix} \begin{bmatrix} \mathbf{u}_i^\top \\ \mathtt{B}_i^\top \end{bmatrix} &= \mathtt{I}_{2\times 2} \end{aligned} \tag{3.8}$$

where, without loss of generality, we have implicitly assumed that the axis of rotation is aligned

with the x-axis of the first object. Thus we can write $\mathtt{S}$ as:

$$\mathtt{S} = \begin{bmatrix} x_1^{(1)} & \cdots & x_{p_1}^{(1)} & x_1^{(2)} & \cdots & x_{p_2}^{(2)} \\ y_1^{(1)} & \cdots & y_{p_1}^{(1)} & 0 & \cdots & 0 \\ z_1^{(1)} & \cdots & z_{p_1}^{(1)} & 0 & \cdots & 0 \\ 0 & \cdots & 0 & y_1^{(2)} & \cdots & y_{p_2}^{(2)} \\ 0 & \cdots & 0 & z_1^{(2)} & \cdots & z_{p_2}^{(2)} \end{bmatrix} \tag{3.9}$$

where now $\mathtt{S}$ is a $5 \times p$ matrix and $p = p_1 + p_2$ (we assume the shapes have been registered to the respective object centroids). Therefore, in the case of a hinge joint the rank of the measurement matrix is at most 5.

3.3 Metric Upgrade

The classic approach in factorisation is to exploit the rank constraint to factorise the measurement matrix into an initial affine solution with a motion matrix $\tilde{\mathtt{M}}$ and a shape matrix $\tilde{\mathtt{S}}$ by truncating the SVD of $\mathtt{W}$ to the rank r specific to the problem. However, this factorisation is not unique since any invertible $r \times r$ matrix $\mathtt{Q}$ can be inserted, leading to the alternative factorisation: $\mathtt{W} = (\tilde{\mathtt{M}}\mathtt{Q})(\mathtt{Q}^{-1}\tilde{\mathtt{S}})$. The problem is to find the transformation matrix $\mathtt{Q}$ that removes the affine ambiguity, upgrading the reconstruction to metric and constraining the motion matrices to lie on the appropriate *motion manifold*.

While in the rigid case the matrix $\mathtt{Q}$ can be explicitly computed linearly by imposing orthonormality constraints on the rows of the motion matrix as shown by Tomasi and Kanade [110], in the non-rigid and articulated cases the metric constraints on the motion matrices are non-linear. Although some closed-form solutions have been recently proposed (see Xiao and Kanade, Hartley and Vidal [130, 128, 49]) these algorithms perform poorly in the presence of noise and cannot cope with missing data. Iterative solutions provide a viable alternative in the presence of noise and missing data and this procedure will be adopted in our proposed algorithm. The factorisation of $\mathtt{W}$ is solved by alternating least-squares where at each step (t) the motion $\mathtt{M}^{(t)}$ and shape $\mathtt{S}^{(t)}$

matrices are optimised separately keeping the other one fixed, as shown in Algorithm 1. This strategy is not uncommon in optimisation problems for SfM (See Buchanan *et al.* [17] for a review). However it is important to note that, differently from previous optimisation schemes, we use a projection step which provides a solution that satisfies the metric constraints exactly. The metric constraints consist of two parts: imposing the correct block structure to the motion matrix and constraining the transpose of the orthographic camera matrices to lie on the Stiefel manifold. In our approach, we impose both constraints simultaneously projecting the motion matrix optimally onto the appropriate motion manifold. As already noticed by Marques and Costeira [70] for the rigid case, these projections not only provide camera matrices which exactly comply with the projection model but also are generally robust to missing and degenerate data.

Algorithm 1 Iterative metric upgrade via alternation for deformable and articulated shape. At each step of the iteration, the motion matrix estimated via least squares is projected onto the motion manifold.

Require: An initial estimate $\mathtt{M}^{(0)}$.

Ensure: A factorisation of $\mathtt{W}$ that satisfies the given metric constraints.

1: Project each frame of $\mathtt{M}^{(t)}$ onto the *motion manifold* of the motion matrices (See Section 3.3.1 for the deformable case and Section 3.3.4 for the articulated case).
2: Estimate $\mathtt{S}^{(t)}$ from the projected $\mathtt{M}^{(t)}$ as: $\mathtt{S}^{(t)} = {\mathtt{M}^{(t)}}^{\dagger}\mathtt{W}$ (where the symbol † indicates the Moore–Penrose pseudo-inverse.
3: Estimate $\mathtt{M}^{(t+1)}$ such that: $\mathtt{M}^{(t+1)} = \mathtt{W}{\mathtt{S}^{(t)}}^{\dagger}$.
4: Repeat until convergence.

Crucially, Step 1 represents the real and novel contribution of this algorithm: an optimisation method which computes the projection of the affine motion components onto the *motion manifold* in which the exact metric constraints are satisfied. Although this problem is non-convex we propose tight convex relaxations (in the sense that the relaxations turned out to be exact in our numerical simulations) that transform the problems into semi-definite (SDP) or second-order cone (SOCP) programs. Steps 2 and 3 alternate the estimation of $\mathtt{M}^{(t)}$and $\mathtt{S}^{(t)}$ assuming the other one known.

Previous approaches have also used iterative methods to perform the metric upgrade in the case

of non-rigid structure including the trilinear alternating least-squares by Torresani *et al*. [112] and by Wang *et al*. [123]. However, even though Torresani *et al*.'s method imposes exact metric constraints on the camera matrices by parametrisation, the update of the camera matrix relies on the assumption that the current estimate differs from the next one only by small rotations. Moreover, the recovery of camera matrices is not optimal. In our case we have an optimal solution to the projection step, which re-estimates the camera matrices and the coefficients to obtain the closest matrix that satisfies the metric constraints. The metric projection algorithm can be visualised in Figure 3.1. After each projection, the shape is recovered via linear least squares. Then we fix the shape to recover a new estimate of the motion matrix. This new estimate will not satisfy the metric constraints, hence a new projection is needed. We iterate until convergence as shown in Figure 3.1. Also Wang *et al*. [123] adopt a trilinear approach where the constraints on the orthographic camera matrices at each frame are imposed using a projection. Their projector is in fact equivalent to the one developed in parallel by Marques and Costeira [70] for rigid shape in the scaled orthographic case. The projection is computed

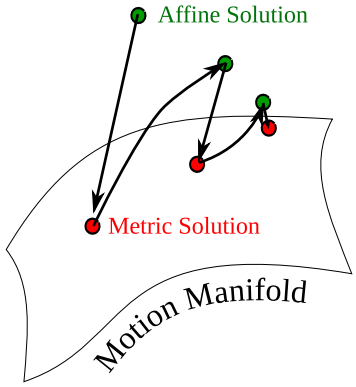


Figure 3.1: Iterative scheme: at each step of the iteration, the motion matrix computed via least squares is projected onto the motion manifold of metric constraints. The process is iterated until convergence

as: $\mathtt{M}_i \mapsto \mathtt{R}_i = \alpha \mathtt{U}\mathtt{V}^\top$ where α is given by the mean of the two singular values $\dfrac{\sigma_1(\mathtt{M}_i)+\sigma_2(\mathtt{M}_i)}{2}$ obtained from the SVD of $\mathtt{M}_i$ (i.e. $\mathtt{M}_i = \mathtt{U}\mathtt{D}\mathtt{V}^\top$). In order to extend such procedure to non-rigid shapes, we first need to define the *motion manifold* for the deformable and articulated cases and

to provide the computational tools to project the motion matrices exactly from affine to metric space.

While other works have chosen to use priors on the shape to constrain the solution to the optimisation problem and obtain the metric upgrade [8, 111, 36], in this work we provide a metric upgrade step that solves an unconstrained least-squares problem and optimally projects the solution onto the *motion manifold* (i.e, computes the closest matrix in the motion manifold with respect to the Frobenius norm). In such regard, we postulate that reliable solutions to the NRSfM problem can be obtained without the use of prior information about the motion of the object or the smoothness of its deformations. In the case of articulated structure, we solve globally for both the motion components related to the bodies and the joint axis with a similar procedure. We now give details on how these projections are computed and the theoretical insights for the *motion manifold* of deformable and articulated shapes.

3.3.1 Metric Projection: Deformable Case

The projection is carried out on each $2 \times 3k$ sub-matrix $\mathtt{M}_i$ as defined in Section 3.2 and it corresponds to solving the following minimisation problem at each frame:

$$\min_{\mathtt{R}_i, l_{i1}\dots l_{ik}} \|\mathtt{M}_i - [l_{i1}\mathtt{R}_i|\dots|l_{ik}\mathtt{R}_i]\|_F^2 \tag{3.10}$$

with the added constraint that $\mathtt{R}_i$ be a 2×3 matrix with orthonormal rows (i.e. $\mathtt{R}_i\mathtt{R}_i^\top = \mathtt{I}_{2\times 2}$). This is equivalent to minimising separately all the 2×3 blocks of $\mathtt{M}_i$ giving:

$$\min_{\mathtt{R}_i} \sum_{d=1}^{k} \min_{l_{i1}\dots l_{ik}} \|\mathtt{M}_{id} - l_{id}\mathtt{R}_i\|_F^2 \tag{3.11}$$

which is equivalent to:

$$\min_{\mathtt{R}_i, l_{i1}\dots l_{ik}} \sum_{d=1}^{k} \|\mathtt{M}_{id}\|_F^2 + l_{id}^2 \|\mathtt{R}_i\|_F^2 - 2l_{id}\,\mathrm{Tr}[\mathtt{M}_{id}^\top \mathtt{R}_i]. \tag{3.12}$$

We can then reformulate the problem by computing the minimum first for l_d (i.e. solving for the zeros of the derivative of eq. (3.12)) given $\mathtt{R}$. This resolves in computing the minimum of the quadratic function in l_d given by $f(l_d) = a\,l_d^2 - 2\,b\,l_d + c$. Such minimum is found in $l_d = b/a$ giving in our case that:

$$l_{id} = \frac{\mathrm{Tr}[\mathtt{M}_{id}^\top \mathtt{R}_i]}{\|\mathtt{R}_i\|_F^2} = \frac{1}{2}\mathrm{Tr}[\mathtt{M}_{id}^\top \mathtt{R}_i]. \tag{3.13}$$

Putting this value back in eq. (3.12) and following with the simplification, the minimisation can be written as:

$$\begin{aligned} &\min_{\mathtt{R}_i} \quad \mathbf{r}_i^\top \left[-\textstyle\sum_{d=1}^{k} \mathbf{m}_{id}\mathbf{m}_{id}^\top\right] \mathbf{r}_i \\ &\text{such that} \quad \mathtt{R}_i\mathtt{R}_i^\top = \mathtt{I}_{2\times 2} \end{aligned} \tag{3.14}$$

where $\mathbf{r}_i = \mathrm{vec}(\mathtt{R}_i^\top)$ and $\mathbf{m}_{id} = \mathrm{vec}(\mathtt{M}_{id}^\top)$. Therefore, this quadratic minimisation problem presents a non-convex constraint given by $\mathtt{R}_i$. In Appendix A we show that it is possible to derive an efficient convex relaxation of the of the constraint set. This set is defined only by linear matrix inequalities (LMI). Therefore the optimisation problem is a Semi-Definite Program (SDP) which can be solved using SeDuMi [103]. Further details, including a proof of the relaxation can be found in [38].

The computed *Stiefel matrix* $\mathtt{R}_i^\top$ is then used to recover the weights l_{id}, obtaining a full non-rigid motion matrix that satisfies the metric constraints. This allows us to solve iteratively for the motion and shape as described in Algorithm 1. This optimal metric projection step is the key to our reconstruction algorithm. In section 3.3.2 we show a tight convex relaxation of this problem that allows us to obtain the global optimum for rotation and deformation weights. The disadvantage of this approach is that the computational complexity of solving a quadratic minimisation problem for each frame in the sequence is too onerous. Each minimisation takes about 2 seconds using SeDuMi toolbox (on a Athlon X2 processor running at 2.6GHz), therefore a sequence of 120 frames would take around 4 minutes to process. While this computation time is not unreasonable for a batch process, in Section 3.3.3 we present a new algorithm based on a Newton optimisation method on the Stiefel manifold to speed up the computation by a factor of

around 130. First we describe the initialisation to the minimisation.

Initialisation for the deformable case

Algorithm 1 requires an initial estimate of the motion matrix $\mathtt{M}_i$ at each frame. This in turn requires initial estimates for the camera matrices $\bar{\mathtt{R}}_i$ and the configuration weights $\bar{l}_{id}$. The rigid motion $\bar{\mathtt{R}}_i$ and the first basis shape $\bar{\mathtt{S}}_1$ are initialised from a rank 3 rigid factorisation of the measurement matrix. The second component of the shape bases is estimated from the residual

$$\mathtt{W}_r = \mathtt{W} - \overline{\mathtt{M}\mathtt{S}}_1 \tag{3.15}$$

A new rank 3 factorisation is performed on $\mathtt{W}_r$ and the new configuration weights l_{i2} can be estimated solving for $l_{i2}\bar{\mathtt{R}}_i = \mathtt{M}_{i2}$ keeping the rotations fixed. This can be solved in a simple way by taking advantage of the orthonormality of $\mathtt{R}$:

$$\begin{aligned} \mathrm{vec}(\mathtt{R}_i)l_{ij} &= \mathrm{vec}(\mathtt{M}_{ij}) \\ \mathrm{vec}(\mathtt{R}_i)^\top \mathrm{vec}(\mathtt{R}_i)l_{ij} &= \mathrm{vec}(\mathtt{R}_i)^\top \mathrm{vec}(\mathtt{M}_{ij}) \\ ||\mathtt{R}||_F^2 l_{ij} &= \mathrm{vec}(\mathtt{R}_i)^\top \mathrm{vec}(\mathtt{M}_{ij}) \\ 2l_{ij} &= \mathrm{vec}(\mathtt{R}_i)^\top \mathrm{vec}(\mathtt{M}_{ij}) \end{aligned}$$

This process is repeated to obtain all k deformation modes. The first rigid factorisation needs full data to give a solution, so we use Marques and Costeira's rigid factorisation algorithm [70] if missing data are present.

3.3.2 Convex relaxation

We have shown in the previous section that finding the optimal projection from the affine solution to the manifold of metric solutions can be re-conduced to solving the following minimisation problem;

$$\min_{\mathtt{R}_i} \quad \mathbf{r}_i^T \left[-\sum_{d=1}^{k} \mathbf{m}_{id}\mathbf{m}_{id}^T \right] \mathbf{r}_i \tag{3.16}$$

where $\mathbf{r}_i = vec(\mathtt{R}_i^T)$ with $\mathtt{R}_i\mathtt{R}_i^T = \mathtt{I}_{2\times 2}$ and $\mathbf{m}_{id} = vec(\mathtt{M}_{id}^T)$. This quadratic minimisation problem presents non-convex constraints given by $\mathtt{R}_i$. Appendix A shows that it is possible to obtain a tight convex relaxation which can be efficiently solved using SeDuMi [103]. Further details can also be found in the technical report by Dodig *et al*. [38]. The computed Stiefel matrix $\mathtt{R}_i$ is then used to recover the weights l_{id}, obtaining a full non-rigid motion matrix that satisfies the metric constraints. This allows us to solve iteratively for the motion and shape as described in Algorithm 1.

3.3.3 Newton method on the Stiefel manifold

The approach described in the previous section will provide an optimal projection onto the *motion manifold* of deformable structure. The first observation we made is that the motion matrix for one frame is not unrelated to the next one. For most common image sequences the motion of the camera is smooth, thus each motion matrix $\mathtt{M}_i$ will not vary much from frame to frame. Therefore, it is not unrealistic to assume that the camera pose at frame i is a good initialisation for an iterative algorithm which tries to compute the pose in the next frame $i+1$. This *warm-start* strategy is not explicitly designed for standard solvers for convex optimisation problems ([103]). Instead, we have adopted a Newton-like iterative optimisation algorithm based on the work of Edelman, Arias and Smith [39]. We can perform optimisation directly on the Stiefel manifold which, for the case of smoothly varying camera poses, will converge locally to the minimum. Of course we lose the optimality of the convex relaxation algorithm. However, empirically we found that in all our experiments with ground truth data, in absence of noise, both algorithms converged to the same minimum.

We now provide additional details on how to compute the Newton step update for the *motion manifold* of deforming shapes. To adhere to the notation in [39] we define the problem as that of minimising a function $\mathtt{F}(\mathtt{Y})$, where $\mathtt{Y}$ is constrained to the set of matrices such that $\mathtt{Y}^\top\mathtt{Y} = \mathtt{I}_{[2\times 2]}$ i.e. it is a *Stiefel matrix*. In our metric projection method, $\mathtt{Y}$ is the $[3\times 2]$ transpose of the camera matrix. The current estimate of the Stiefel matrix is updated using the geodesic formula

for a unit step $t = 1$

$$\mathtt{Y}(t) = \mathtt{YM}(t) + \mathtt{QN}(t) \tag{3.17}$$

In order for the update to move along the geodesic, the $[2 \times 2]$ matrices $\mathtt{M}(t)$ and $\mathtt{N}(t)$ in 3.17 are given by the matrix exponential

$$\begin{pmatrix} \mathtt{M}(t) \\ \mathtt{N}(t) \end{pmatrix} = \exp t \begin{pmatrix} \mathtt{Y}^\top \Delta & -\mathtt{R}^\top \\ \mathtt{R} & 0 \end{pmatrix} \begin{pmatrix} \mathtt{I}_{[2\times 2]} \\ 0 \end{pmatrix} \tag{3.18}$$

Given the Newton direction Δ, matrices $\mathtt{Q}_{[3\times 2]}$ and $\mathtt{R}_{[2\times 2]}$ in 3.17, 3.18 are given by the compact QR-decomposition of $(\mathtt{I}_{[3\times 3]} - \mathtt{YY}^\top)\Delta$.

Δ is the $[3 \times 2]$ matrix defined by the equation

$$\Delta = -\text{Hessian}^{-1}(\mathtt{F}_Y - \mathtt{YF}_Y^\top \mathtt{Y}) \tag{3.19}$$

Where $\mathtt{F}_Y$ is a $[3 \times 2]$ matrix of first derivatives of the function $\mathtt{F}$ with respect to the elements of $\mathtt{Y}$, and the Hessian is the $[3 \times 3]$ matrix of second derivatives of the cost function $\mathtt{F}$ with respect to the three degrees of freedom in the manifold.

We apply the iterative Newton method (more theoretical insights can be found in [39]) to the cost function given by equation (3.14), using the solution to the previous frame as an initialisation. Evidently, the first frame has to be solved with the previously proposed convex relaxation. In our experiments this new solution provided a remarkable speed-up, solving the whole factorisation problem about 130 times faster than the original method, without losing optimality as observed in the experimental trials. Notice that in this case the assumption that the camera pose varies smoothly is just an initialisation strategy and not a prior term in our minimisation. Our smoothness assumption does not add an explicit penalty term to the cost function to penalise strong deformations or camera motions as other authors do [8, 111].

3.3.4 Metric Projection: Articulated Case

Projection onto the *motion manifold* of the universal joint can be simply solved by performing two separate rigid factorisations for each of the parts of the articulated object followed by an estimation of the joint location as presented by Tresadern and Reid [113]. The hinge joint is far more interesting given the non-linear relations between the motion subspaces. The two objects cannot be reconstructed independently, for each reconstruction is subject to reconstruction ambiguities arising from orthographic projection (chirality and average depth). The two objects must be reconstructed jointly, in order to recover the hinge joint. Is is shown in Yan and Pollefeys results [131] that two rigid bodies coupled by a hinge joint will result in tracking data of lower dimensionality than two independent rigid bodies. In this work we are going to adopt the same formulation defined by Tresadern and Reid [113], who propose a factorisation approach. We can apply our algorithm to solve the difficult problem of metric upgrade. Instead of looking for a linear solution, we can apply the metric projections algorithm to recover motion and shape matrices. In the articulated case the projection problem is to find a matrix that satisfies the constraints given by a rotation axis. Following eq. (3.6) the projection problem for the hinge *motion manifold* can be written at each frame as the following minimisation:

$$\min_{\mathbf{u},\mathtt{A},\mathtt{B}} J(\mathbf{u},\mathtt{A},\mathtt{B}) = \|\mathbf{u}-\mathbf{x}\|^2 + \|\mathtt{A}-\mathtt{Y}\|_F^2 + \|\mathtt{B}-\mathtt{Z}\|_F^2 \,, \tag{3.20}$$

subject to the constraints defined in eq. (3.8). Here $\mathbf{x}$, $\mathtt{Y}$ and $\mathtt{Z}$ are obtained directly from the affine motion matrix $\tilde{\mathtt{M}}_i = [\mathbf{x}|\mathtt{Y}|\mathtt{Z}]$, recovered through SVD. Equation (3.20) can be reformulated as the minimisation of $J(\mathbf{u},\mathtt{A},\mathtt{B})$ only as a function of the common axis $\mathbf{u}$ such that:

$$\min_{\mathbf{u},\mathtt{A},\mathtt{B}} J(\mathbf{u},\mathtt{A},\mathtt{B}) = \min_{\mathbf{u}} J(\mathbf{u}). \tag{3.21}$$

This is possible as we will show that, once the optimal $\mathbf{u}$ is estimated, it is straightforward to obtain $\mathtt{A}$ and $\mathtt{B}$ in closed form. The equivalent cost function $J(\mathbf{u})$ can be written as:

$$\min_{\mathbf{u}} J(\mathbf{u}) = \min_{\mathbf{u}} \left\{ \|\mathbf{u}-\mathbf{x}\|^2 + \phi_Y(\mathbf{u}) + \phi_Z(\mathbf{u}) \right\}. \tag{3.22}$$

Thus now we will show how to transform the minimisation of $\|\mathtt{A}-\mathtt{Y}\|_F^2$ into the minimisation of $\phi_Y(\mathbf{u})$ (the same reasoning can be replicated for $\phi_Z(\mathbf{u})$). First, we use the polar decomposition to change variables as $\mathtt{A}=\mathtt{PQ}$ where $\mathtt{P} \succeq 0$ (i.e. $\mathtt{P}$ is a semi-definite matrix) and $\mathtt{Q}$ is orthogonal (both $\mathtt{P}$ and $\mathtt{Q}$ are 2×2). Moreover, given the metric constraints in eq. (3.8), it follows that $\mathtt{P}^2 = \mathtt{I}-\mathbf{u}\mathbf{u}^\top$. Thus, the matrix $\mathtt{I}-\mathbf{u}\mathbf{u}^\top$ must be positive definite, restricting the vector $\mathbf{u}$ to be inside the unitary circle. Then, for a chosen $\mathbf{u}$ we can write $\phi_Y(\mathbf{u})$ as:

$$\begin{aligned}
\phi_Y(\mathbf{u}) &= \min_{\mathtt{QQ}^\top=\mathtt{I}} \left\| (\mathtt{I}-\mathbf{u}\mathbf{u}^\top)^{1/2}\mathtt{Q}-\mathtt{Y} \right\|_F^2 \\
&= \min_{\mathtt{QQ}^\top=\mathtt{I}} \left\{ \left\| (\mathtt{I}-\mathbf{u}\mathbf{u}^\top)^{1/2} \right\|_F^2 + \|\mathtt{Y}\|_F^2 \right. \\
&\qquad \left. -2\,\mathrm{Tr}\left(\mathtt{Y}^\top \left(\mathtt{I}-\mathbf{u}\mathbf{u}^\top\right)^{1/2} \mathtt{Q} \right) \right\}.
\end{aligned}$$

Minimising this cost function over the orthogonal matrix $\mathtt{Q}$ equals to maximising the trace in the previous expression.

Using the property:

$$\max_{\mathtt{QQ}^\top=\mathtt{I}} \{\mathrm{Tr}(\mathtt{XQ})\} = \sigma_1(\mathtt{X}) + \sigma_2(\mathtt{X}) + \cdots + \sigma_n(\mathtt{X}) = \|\mathtt{X}\|_{\mathrm{N}} \tag{3.23}$$

where $\|\mathtt{X}\|_{\mathrm{N}}$ denotes the *nuclear norm* of $\mathtt{X}$ (i.e. the sum of its singular values), we can write that:

$$\phi_Y(\mathbf{u}) = 2 - \|\mathbf{u}\|^2 + \|\mathtt{Y}\|_F^2 - 2\left\| \left(\mathtt{I}-\mathbf{u}\mathbf{u}^\top\right)^{1/2} \mathtt{Y} \right\|_{\mathrm{N}} \tag{3.24}$$

The same reasoning can be replicated for $\phi_Z(\mathbf{u})$ giving the final optimisation problem to be

solved as:

$$\begin{aligned} \min \quad & -\|\mathbf{u}\|^2 - 2\mathbf{u}^\top \mathbf{x} - 2\left\|\left(\mathtt{I} - \mathbf{u}\mathbf{u}^\top\right)^{1/2}\mathtt{Y}\right\|_{\mathrm{N}} \\ \|\mathbf{u}\| \leq 1 \quad & -2\left\|\left(\mathtt{I} - \mathbf{u}\mathbf{u}^\top\right)^{1/2}\mathtt{Z}\right\|_{\mathrm{N}} \end{aligned} \tag{3.25}$$

Once the optimal $\mathbf{u}^*$ is found we substitute back in order to recover the solution for $\mathtt{A}$ (and similarly for $\mathtt{B}$). First we obtain $\mathtt{Q}$ from the SVD of $\mathtt{Y}^\top(\mathtt{I} - \mathbf{u}^*\mathbf{u}^{*\top})^{1/2} \mapsto \mathtt{UDV}^\top$ leading to $\mathtt{Q} = \mathtt{VU}^\top$. The matrix $\mathtt{P}$ is simply given knowing that $\mathtt{P}^2 = \mathtt{I} - \mathbf{u}^*\mathbf{u}^{*\top}$. This will result in the matrix that exactly satisfies the metric structure of a hinge joint. The optimisation of the cost function in eq. (3.25) is not trivial since the cost function is non-convex and non-smooth. However the domain in which the function resides is constrained (i.e. the unitary circle) and the value of eq. (3.25) for an arbitrary $\mathbf{u}$ can be computed efficiently without the need of calculating the nuclear norm at each sample. The optimisation can be then solved with a simple exhaustive search algorithm in which the function sampling can be computed in a small amount of time (this was in fact the strategy used in [85]). The resulting brute-force algorithm is visualised in figure 3.2, we can scatter a uniformly distributed grid of points in the unitary circle, and evaluate the cost function at each point. If a finer grid is required, that can be cast from the minimum found in the coarse grid, as shown. We obtain good results with this simple exhaustive search minimisation, but in the following section we will propose a convex relaxation that will find the optimum in a much shorter time and with greater accuracy.

Convex relaxation for the articulated case

Although the cost function in equation (3.25) is non-convex, in Appendix B we propose an efficient convex relaxation. Differently from the deformable case, the reformulation leads to two cases. As shown in Appendix B, in one case the problem becomes a semi-definite program (SDP) and in the other a second order cone program (SOCP) both of which can be efficiently solved with standard convex optimisation tools [103]. In all of our numerical experiments we found that the proposed convex relaxations were exact, thereby solving indeed (3.25). Compared to the full search method we described in the previous section, this convex optimisation speeds

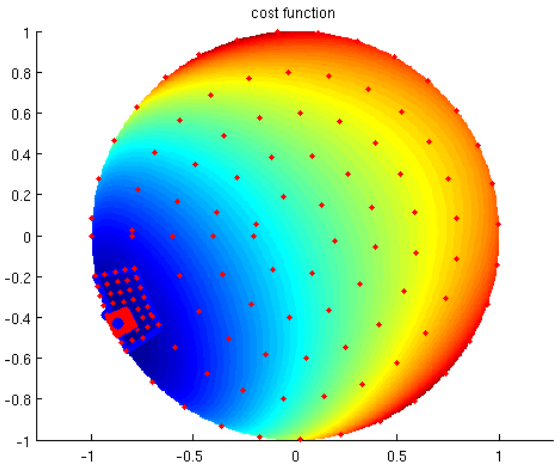


Figure 3.2: Exhaustive search for minimising the cost function 3.25. A coarse grid of candidate points is scattered on the unitary circle, and the process is repeated on a smaller area around the evaluated minimum.

up the computation by a factor of around ten. A second advantage is that we avoid the problem of the accuracy of the solution depending on the density of the interval grid in the parameter space as in the full-search algorithm. The full details of the proposed convex relaxation can be found in Appendix B.

Initialisation for the articulated case

We first consider the two bodies separately and then perform a rigid factorisation for each shape. Given this factorisation, we can then obtain an initial closed form solution for the metric upgrade in the case of a hinge using the linear method by Tresadern and Reid [113].

3.4 Reconstruction with Missing Data

Incomplete image tracks are a common occurrence in SfM tasks and several algorithms have been proposed in order to cope with the missing data problem within the factorisation framework (see Buchanan and Fitzgibbon for a review [17]). Our new factorisation approach presented in the previous section can be modified to account for missing entries in `W`. The strength of our approach lies in the fact that the *motion manifold* constrains the estimated motion of the missing

2D image points since we only allow trajectories that satisfy the metric constraints exactly. Instead of using only the known image tracks to solve for the camera matrices, basis shapes and deformation coefficients as the trilinear least-squares approaches do [112, 123], we opt for an iterative scheme. At each step of the iteration we re-compute the missing entries in the measurement matrix $\mathtt{W}$ using the current estimates of the motion and shape matrices that have been projected onto the correct *motion manifold*. In our experimental validation, reported in Section 3.5, we have found that dealing with missing data using the iterative scheme described here allows to deal with higher percentages of missing data than using only the available data as Wang *et al*. do in their RCPF approach [123]. The steps of this method are summarised in Algorithm 2.

Algorithm 2 Metric Projections algorithm in the presence of missing data.

Require: An initial estimate $\mathtt{W}^{(0)}$ of the missing data in $\mathtt{W}$.
Ensure: A factorisation of $\mathtt{W}$ that satisfies the given metric constraints.
1: Remove the 2D centroid $\mathtt{T}^{(t)}$ from $\mathtt{W}^{(t)}$, i.e. $\hat{\mathtt{W}}^{(t)} = \mathtt{W}^{(t)} - \mathtt{T}^{(t)}$.
2: Factorise $\hat{\mathtt{W}}^{(t)} = \mathtt{M}^{(t)}\mathtt{S}^{(t)}$ using Algorithm 1.
3: Estimate the missing data entries of $\mathtt{W}$ as $\mathtt{W}^{(t+1)} = \mathtt{M}^{(t)}\mathtt{S}^{(t)} + \mathtt{T}^{(t)}$
4: Repeat until convergence.

The algorithm requires an initial estimate of the missing entries in the measurement matrix $\mathtt{W}$. For this purpose, we have used the rigid factorisation algorithm of [70] to obtain an initial rigid fit of the missing entries. In the case of articulated structure we apply the algorithm independently to each of the bodies. The iterations are stopped when the distance $||\mathtt{W}^{(t+1)} - \mathtt{W}^{(t)}||_F$ falls below a user-defined threshold, that is, when the new estimate does not modify the previous values much.

3.5 Experiments

First we show results for the recovery of deformable structure, followed by results for articulated structure. We evaluate the performance of our algorithms quantitatively on various motion capture sequences, for which ground truth was available, and we compare our results with some

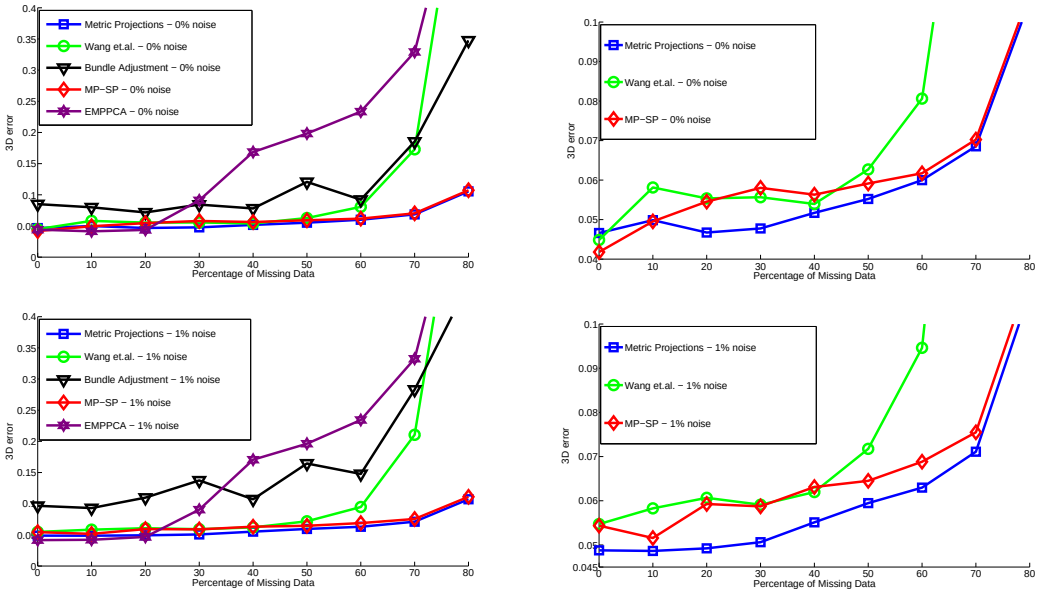


Figure 3.3: Missing data tests on the *Face1* Motion Capture sequence. Plots show the average 3D error over 100 tests for increasing levels of randomly generated missing data. We compare the results obtained with: Metric Projections (MP), EMPPCA, Bundle Adjustment (BA), Rotation Constrained Powerfactorization (RCPF) and MP with a Simple Projector (MP-SP). The plots on the left column show the average 3D errors in the noise-less case (top) and with added Gaussian noise (bottom) of $\sigma = 1\%$. The plots on the right show a zoomed-in version of the three best performing algorithms (MP, RCPF and MP-SP). The performance of MP and MP-SP is similar although MP outperforms MP-SP.

current state of the art NRSfM algorithms [111, 36, 123]. In the case of the articulated Metric Projections (MP) algorithm we evaluated against Tresadern and Reid linear method [113]. Notice that we do not compare with Yan and Pollefeys' approach [133] since their proposed method does not perform a 3D metric reconstruction of the shape and joint axes – only the 2D projection of the axes in the image is computed. Finally we demonstrate our algorithms on real image sequences. We have made our code and sequences available for download on our website[3].

[3] http://www.dcs.qmul.ac.uk/~lourdes/code.html

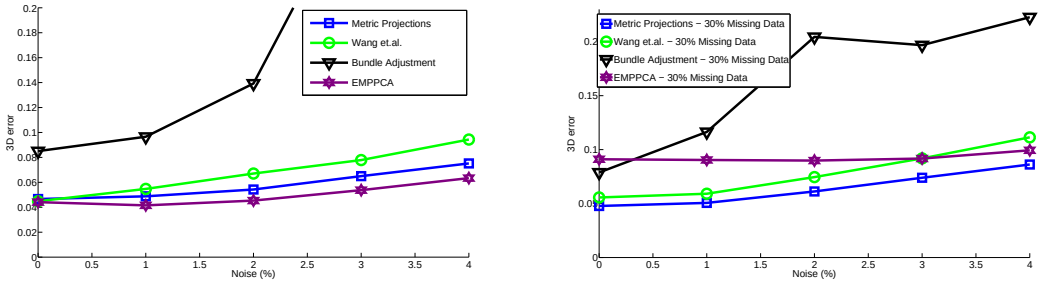


Figure 3.4: Noise test for the *Face1* Motion Capture sequence in the cases of full data case (left) and 30% missing data (right). We show 3D errors versus percentage of added Gaussian noise. In the full data case (left), EMPPCA performs marginally better while in the missing data case (right) MP is the best performing algorithm.

3.5.1 Deformable Structure

Synthetic Experiments – Motion capture data

In our synthetic experiments we used two different 3D motion capture sequences, both showing faces. The first sequence, *Face1*, was captured in our own laboratory using a VICON system tracking a subject wearing 37 markers on the face. The 3D points were then projected synthetically onto an image sequence 74 frames long using an orthographic camera model. The second sequence, *CMU* face sequence[4], is motion capture data made available by Torresani *et al*. [111]. The subject wore 40 markers tracked by a motion capture system and the orthographic projection is performed by simply discarding the third coordinate of each 3D point. Note that although the projection of the ground truth 3D data on the images is synthetic the deformations are realistic since they come from real motion capture sequences. The 2D image data is therefore not synthetic and it contains some noise due to the motion capture estimation errors.

Our proposed Metric Projection algorithm (MP) is tested against various state of the art algorithms: EMPPCA [111], which is currently perceived to be the state of the art/baseline algorithm and for which code has been made available on-line; Rotation Constrained Power Factorisation (RCPF) [123], which is the most closely related approach to our new MP algorithm since it also performs a (rigid) projection of the camera matrices as we described in Section 3.1.1, and a Bun-

[4] http://www.cs.dartmouth.edu/~lorenzo/nrsfm.html

dle Adjustment algorithm (BA) designed for NRSfM [36] where the orthonormality constraint on the rotation matrices is imposed through parametrisation.

In the case of missing data we also report results with a modified version of our Algorithm 2. We are interested in assessing (in the case of missing data) the gain in performance achieved by using our bilinear scheme followed by our new optimal metric projector instead of Wang *et al.*'s trilinear scheme followed by their simpler projector of the camera matrices onto the motion manifold [123]. In order to do this we have designed a new algorithm that we call MP-SP: *Metric Projection with Simple Projection*. The idea is to use our outer loop to deal with the missing data and substitute Step 2 in Algorithm 2 with Wang *et al.*'s RCPF algorithm. In this way we can test an algorithm with the same initialisation, the same iterative outer loop to deal with missing data but using Wang *et al.*'s trilinear approach with the simpler projection step to perform factorisation. Note that this new scheme (MP-SP) is not Wang *et al.*'s RCPF algorithm: the missing data is dealt with in a different way. Effectively, our Algorithm 2 (MP in the case of missing data) and the new MP-SP have exactly the same structure. They only differ in the factorisation algorithm used in Step 2: in the case of Algorithm 2 it is our MP algorithm for full data (Algorithm 1) while in the case of MP-SP it is Wang *et al.*'s RCPF algorithm.

To test the performance of the algorithms we computed the 3D error, which we defined as the Frobenius norm of the difference between the recovered 3D shape $\mathtt{S}$ and the ground truth 3D shape $\mathtt{S}_{GT}$. The error is normalised against the Frobenius norm of the ground truth shape $||\mathtt{S} - \mathtt{S}_{GT}||_F / ||\mathtt{S}_{GT}||_F$. We subtract the centroid of each shape and align them with Procrustes analysis. In the noise tests zero mean additive Gaussian noise was applied with standard deviation $\sigma = \mathrm{n} \times \mathrm{s}/100$ where n is the noise percentage and s is defined as $\max(\mathtt{W})$ in pixels.

Initialisation: EMPPCA was initialised with its own method supplied by the authors in their software [111] (camera matrices and mean shape are computed using Tomasi and Kanade rigid factorisation [110] while deformation basis and coefficients are estimated through iterative PCA of the shape residuals). The mean shape and camera matrices were initialised in an identical way for BA, RCPF, MP and MP-SP using Marques and Costeira's rigid factorisation algorithm

[70] to compute rigid shape and camera matrices. The deformation basis and coefficients in the BA algorithm were initialised in the same way as EMPPCA. Wang *et al*. [123] RCPF algorithm only needs an initialisation for the non-rigid deformation basis which were set to small random values (as indicated by the authors in [123]). Our MP algorithm initialisation only needs the coefficients which were initialised through iterative PCA of the residuals of the measurement matrix `W` as explained in Section 3.3.1. The outer iterative loop of MP and MP-SP algorithms also require an initialisation of the missing data for which the rigid factorisation by Marques and Costeira [70] was used.

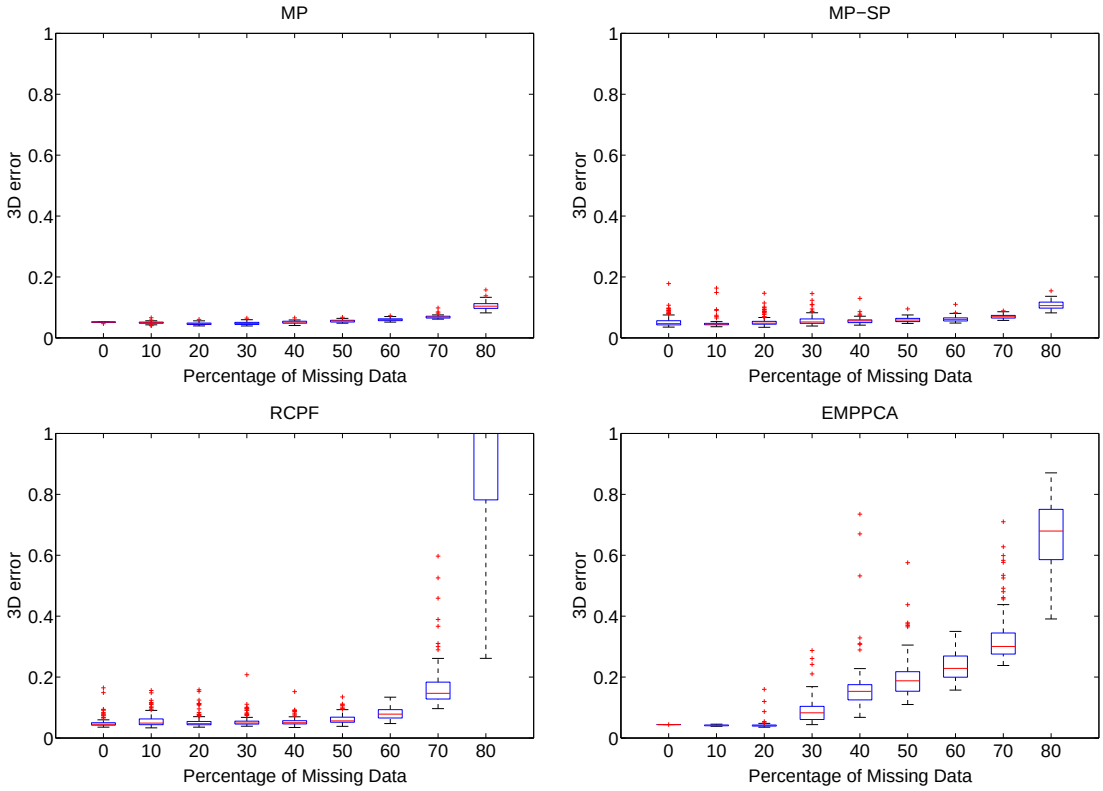


Figure 3.5: Box-plots showing statistics on the distributions of the 3D reconstruction errors for the ”Face1” sequence in the case of no added noise.

Missing data and noise tests

In Figure 3.3 we compare the performance of our new algorithm MP with EMPPCA, RCPF, BA and MP-SP for the *Face1* sequence in the case of increasing levels of missing data ranging from 10% to 80%, generated by deleting entries from the measurement matrix randomly. For each level of missing data we averaged the results of 100 runs varying the missing data mask. Tests in which the 3D error was higher than 100% were considered as outliers and were not used to compute the average. In all experiments the number of basis shapes was fixed to $k = 5$.

The top row of Figure 3.3 shows the results in the noiseless case, while the bottom row shows the results in the more realistic case of 1% image noise. The plots in the left column show the 3D error of all the algorithms (MP, EMPPCA, RCPF, BA and MP-SP) while the plots on the right column show a zoomed-in version for the algorithms showing the best performance (MP, MP-SP and RCPF), which interestingly, enforce orthonormality constraints on the camera matrices through projection. The left plots in the noiseless (top) and 1% noise case (bottom) show that EMPPCA and BA are the worse performing algorithms in the presence of missing data. EMPPCA can cope with up to 20% missing data before the error starts to grow steadily. BA gives the highest 3D errors for low ratios of missing data but appears to show more resilience to higher ratios of missing data than EMPPCA. However, it also breaks down after 50% missing data.

The plots in the right column of Figure 3.3 show a zoomed-in view of the best performing algorithms. Our new MP algorithm achieves the smallest overall 3D errors both in the noiseless case (right-top) and more clearly in the 1% noise test (right-bottom). RCPF [123] shows good performance until levels of around 50% missing data but the errors grow quickly after that. The second best performing algorithm is MP-SP which uses our outer loop to deal with missing data and RCPF internally to perform factorisation. Although its performance is comparable to MP, the 3D error curve for MP lies below – for instance in the 1% noise case (bottom-right)the 3D reconstructions obtained with MP are on average around 1% better than with MP-SP.

It is worth discussing three interesting facts revealed by the results of these tests for increasing

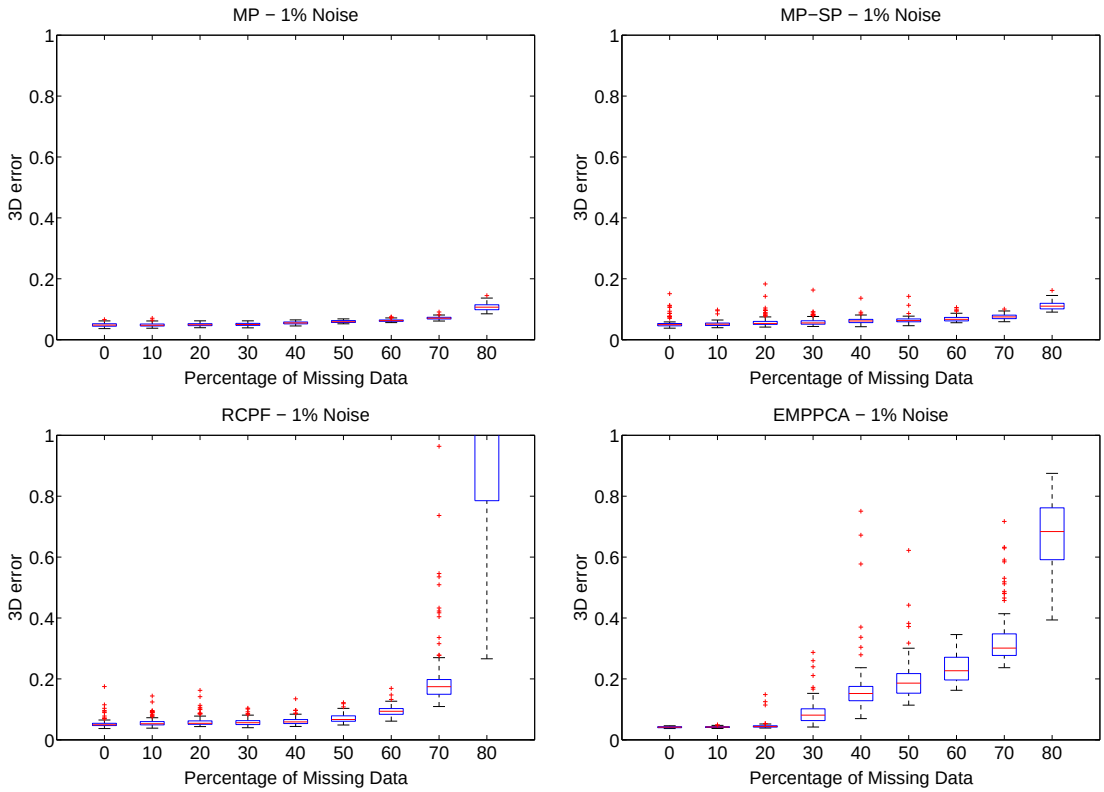


Figure 3.6: Box-plots for the 1% noise test showing statistics on the distributions of the 3D reconstruction errors for the ”Face1” sequence.

levels of missing data. First, all top three performing algorithms (MP, MP-SP and RCPF) include a projection step of the camera matrices to deal with metric constraints. BA and EMPPCA, on the other hand, impose the orthonormality constraints through parametrisation (quaternions in the case of BA and exponential map in the case of EMPPCA). Secondly, while RCPF, MP-SP and MP show very similar performance for missing data ratios of up to 50%, for higher ratios MP-SP and MP greatly outperform RCPF. The only difference between MP-SP and RCPF is the way in which they deal with missing data: RCPF uses only the known 2D image tracks while MP-SP uses an outer loop to re-estimate the missing data at each step of the iteration. Note that they were both initialised in the same way as MP. Finally, the performance of MP is about 1% better than MP-SP. However, MP-SP runs around 25% faster (see Figure 3.8 for algorithm run-times). Therefore if run-time is an issue MP-SP could be used instead of MP

Figure 3.7: Structured missing data mask used for the experiment described in Section 3.5.1. Each column is a point track, points in black are marked as visible, points in white are marked as occluded.

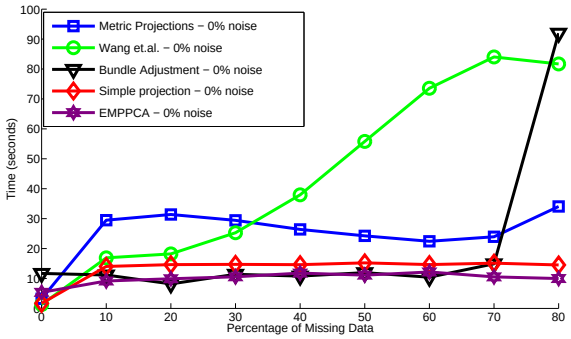


Figure 3.8: Comparison of run-times (in seconds) averaged over 100 tests, versus percentage of missing data. Tests were performed using a 4-core Xeon processor running at 2.8GHz, with 24GB of RAM.

without compromising performance too much but of course improved results would be achieved with MP.

In Figure 3.4 we show comparative noise tests for EMPPCA, BA, RCPF and MP in the case of full data (left) and 30% missing data (right). We show results for noise levels of up to 4% meaning that the value of the variance σ is up to 4% of the size of the object in the image. It is clear that BA, is the most vulnerable algorithm to noise in the image coordinates. Note also that EMPPCA, RCPF and MP perform very similarly with EMPPCA performing slightly better in the full data case and MP in the 30% missing data case. The results were averaged over 100 runs.

To show more details on the distribution of 3D errors in the results, Figure 3.5 reports box-plots of the errors for all the trials performed in the case of 0% noise. In the noiseless case, the only difference between trials is the missing data visibility matrix (randomly generated). In Figure 3.6 we show box-plots of the errors in the 1% noise case. Overall, our MP method was the one that obtained the most robust statistics.

Figure 3.9 shows front and side views of the 3D reconstruction results for one of the runs of the *Face1* sequence with no noise and 40% missing data. The top row shows some frames of the motion capture session (which do not correspond to the reconstructed ones below), the second, third and fourth rows show ground truth values and 3D reconstruction results obtained with our method MP, EM-PPCA and RCPF respectively. Our reconstruction is closer to the ground truth shape. The average 3D reconstruction error over all the frames of this sequence was 4.7% with MP, 13.1% with EMPPCA and 9.0% with RCPF.

Figure 3.10 compares ground truth with the results obtained with MP, EMPPCA and RCPF for the *CMU* face sequence with full data and with 30% missing data. In the full data case all algorithms perform similarly. However, in the missing data case, our algorithm recovers the 3D shape correctly and outperforms Torresani *et al*.'s. The 3D errors against ground truth motion capture data were the same for RCPF and MP (2%), both for full data and 30% missing data, while for EMPPCA the 3D error is low (1.8%) in the full data case, but very high (35%) in the missing data case.

Figure 3.8 shows the mean run-times expressed in seconds, for the experiment in Figure 3.3, for EMPPCA, BA, RCPF and MP for different ratios of missing data. Tests were performed using a 4-core Xeon processor running at 2.8GHz, with 24GB of RAM. All implementations are in MATLAB. The fastest algorithms are BA and EMPPCA. However the code for BA and EMPPCA provided by the authors contains some parts of optimised MEX code. At the expense of losing some accuracy, as we saw in Figure 3.3, MP-SP runs around 30% faster than MP since the projection step is much more simple. Note that RCPF requires a large number of iterations in order to achieve convergence after 30% missing data. Therefore, adding the outer loop to RCPF

to deal with missing data as we did in MP-SP improves the convergence in this case.

Each of the tested approaches uses its own custom initialisation for the optimisation routines. This difference is dictated by the fact that each method starts the iterations from a different parameter set. While all algorithms require an initial estimate for the camera matrices and the mean rigid shape, for instance, our MP requires an initial estimate of the motion matrix `M`, BA and EMPPCA need a first guess of the basis shapes and deformation weights while RCPF requires an initial estimate of basis shapes. Since each initialisation is inherited from the specific structure of the method, evaluating each approach with exactly the same initialisation is not feasible. However, we have attempted to make the initialisations as uniform as possible by using Marques and Costeira [70], which fills in the missing entries in the data matrix, to initialise the mean shape and camera projection matrices in the case of MP, BA and RCPF. Note that only our algorithm, MP, uses the missing entries explicitly in the outer loop proposed in Algorithm 2, while BA and RCPF only use the known data in the estimation.

Synthetic Experiments – Structured occlusions

While it is important to conduct experiments with randomly generated missing data to control its percentage in the simulation, we also performed a test with a missing data mask where points are occluded in a structured way, as it would happen for instance due to self-occlusions.

In order to generate a more realistic missing data pattern we have computed surface normals from the sparse 3D motion capture data using the *taglut* algorithm[5].The computed angles between surface normal and camera viewing direction for all frames have been thresholded at 60 degrees, marking large angles as occluded. Although the knowledge of surface normals allows to simulate self-occlusions, the strong sparseness of the measured points does not permit to simulate realistic self-occlusions. However, the resulting occlusion pattern is structured and not random as in the previous tests. The resulting occlusion mask is shown in Figure 3.7 – the amount of missing data resulting from this computation was 32%. The resulting visibility matrix captures well the structured disappearance of image features. We then ran our MP Algorithm 2 on the input 2D

[5] http://jmfavreau.info/?q=en/taglut

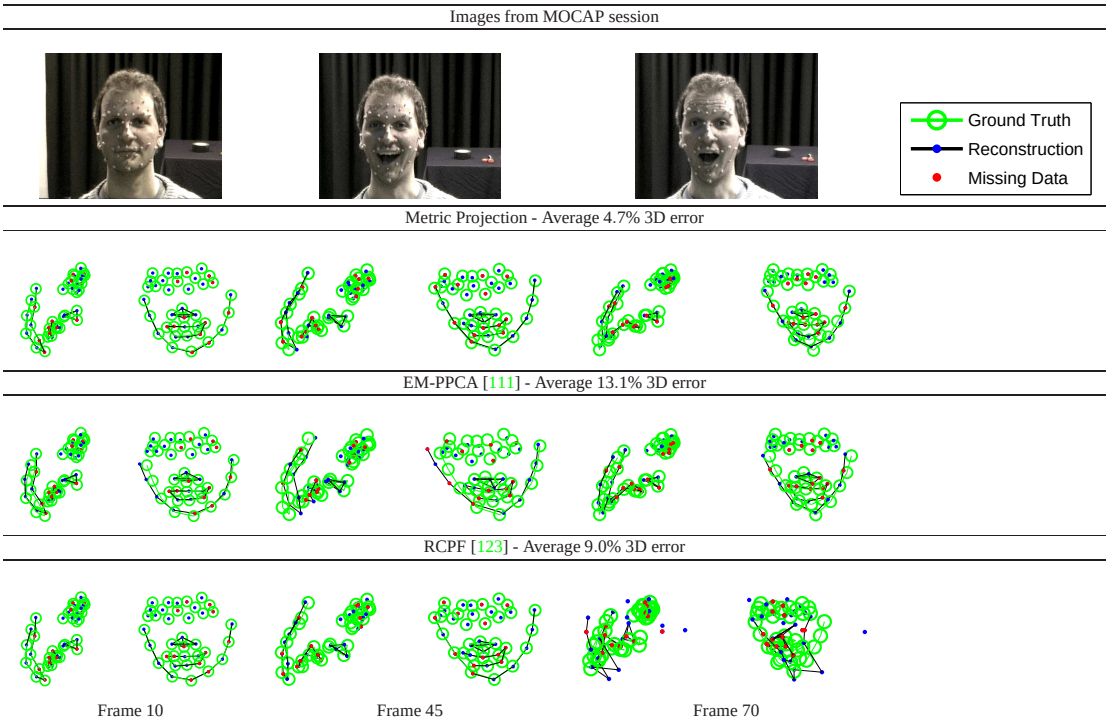


Figure 3.9: 3D reconstruction results for a single run of the the *Face1* motion capture sequence with 40% missing data. Missing points are highlighted in red. Top row: Some frames of the original motion capture take (the images do not correspond to the reconstructed frames shown below). Second, third and fourth rows: side and front views for some frames of the 3D reconstruction for our Metric Projection method, Torresani *et al.*'s EM-PPCA and Rotation Constrained Power Factorisation. We show ground truth (green circles) and reconstructed points (blue dots if visible red if not). The wire-frame lines are only shown for visualisation purposes.

data, obtaining a 3D reconstruction error of 5.4%. A visual comparison of the reconstructed 3D against ground truth motion capture data is given in Figure 3.11 and Figure 3.12. We also compare this result with other techniques, and show that MP outperforms other methods in this case. In particular, EMPPCA [111] obtains 8.6% 3D reconstruction error, and Wang *et al.*'s RCPF [123] achieves 8.4% error. This test shows that the advantage of metric projection remains even when the occluded points are not selected randomly, but in the more realistic case of structured occlusions.

Real Sequences

Cushion Sequence

In our first experiment we tested our algorithm on an image sequence of a cushion bending and stretching, in which 90 points were tracked manually. The results are shown in Figure 3.13. Our algorithm reconstructs successfully the 3D point cloud and its deformations. We used this data to generate a texture-mapped view of the reconstructed object. We also performed a quantitative evaluation by comparing the 3D reconstruction obtained with full data to those obtained with different percentages of missing data – generated by deleting randomly entries on the measurement matrix. The difference (computed in the same way as we compute the 3D error) between the 3D shape reconstructed with full data and the shapes obtained with 10%, 20% and 30% missing data are 3.8%, 5.7% and 5.9% respectively . We also measured the average image reprojection error which was 0.1 pixels with full data, and 1.1, 1.2 and 1.4 pixels for the 10%,20% and 30% missing data cases respectively. In Figure 3.14 we show the 3D reconstruction results on the cushion sequence with 10% missing data generated randomly.

Franck Sequence

We also used the Franck sequence[6] taken from a video of a person engaged in conversation. We selected 700 frames from the 5000 frame sequence. An Active Appearance Model (AAM) was used to track 68 features on the face. Figure 3.15 shows three frames of the original images and a view of the resulting 3D reconstruction in the cases of complete 2D data (second row) and 20% missing data (third row). We also show the 3D reconstruction achieved with EMPPCA for the full data case as a baseline (fourth row). However, we could not show the results for EMPPCA for 20% missing data since already for that value, the errors were too high and the reconstruction was meaningless. The last two rows (fifth and sixth) show the results achieved with the RCPF algorithm in the cases of full data and 20% missing data. The number of basis shapes was chosen to be 6 in this experiment. Our algorithm appears to achieve the best 3D reconstructions in this real sequence with and without missing data.

[6] www-prima.inrialpes.fr/FGnet/data/01-TalkingFace/talking_face.html

Figure 3.10: 3D reconstruction results for the "CMU" face motion capture sequence. First row: input 2D data. Second and third rows: full data results, Metric Projection and EM-PPCA. Reconstruction (blue dots) are compared with ground truth data (green circles). Fourth, fifth and sixth rows: Results for 30% missing data (highlighted in red).

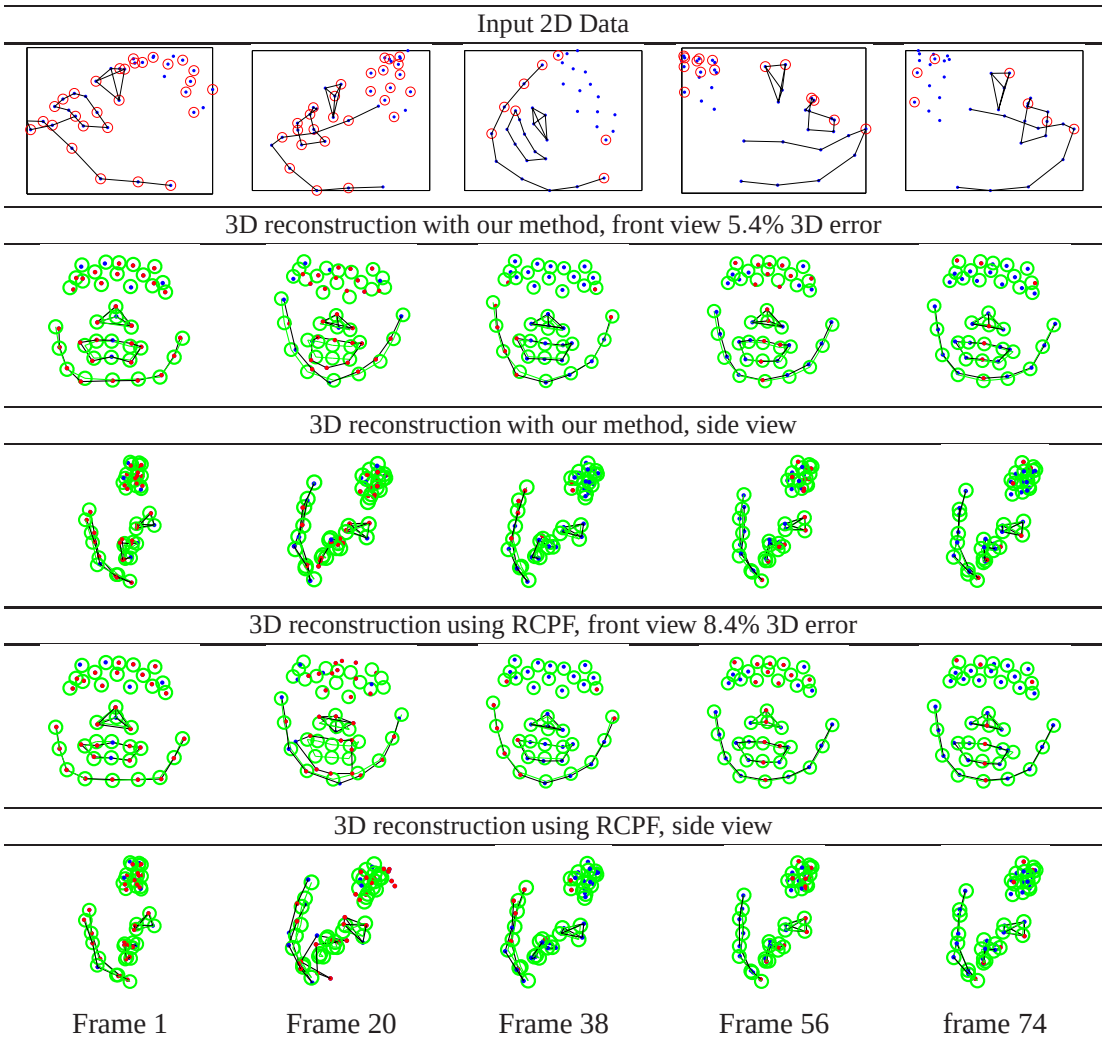


Figure 3.11: 3D reconstruction results obtained for the *Face1* motion capture sequence with the structured missing data mask shown in Figure 3.7. Top row: 2D input data. Comparison between our MP algorithm (second and third rows) against RCPF (fourth and fifth rows). Ground truth shown in green, missing data points highlighted with a red circle.

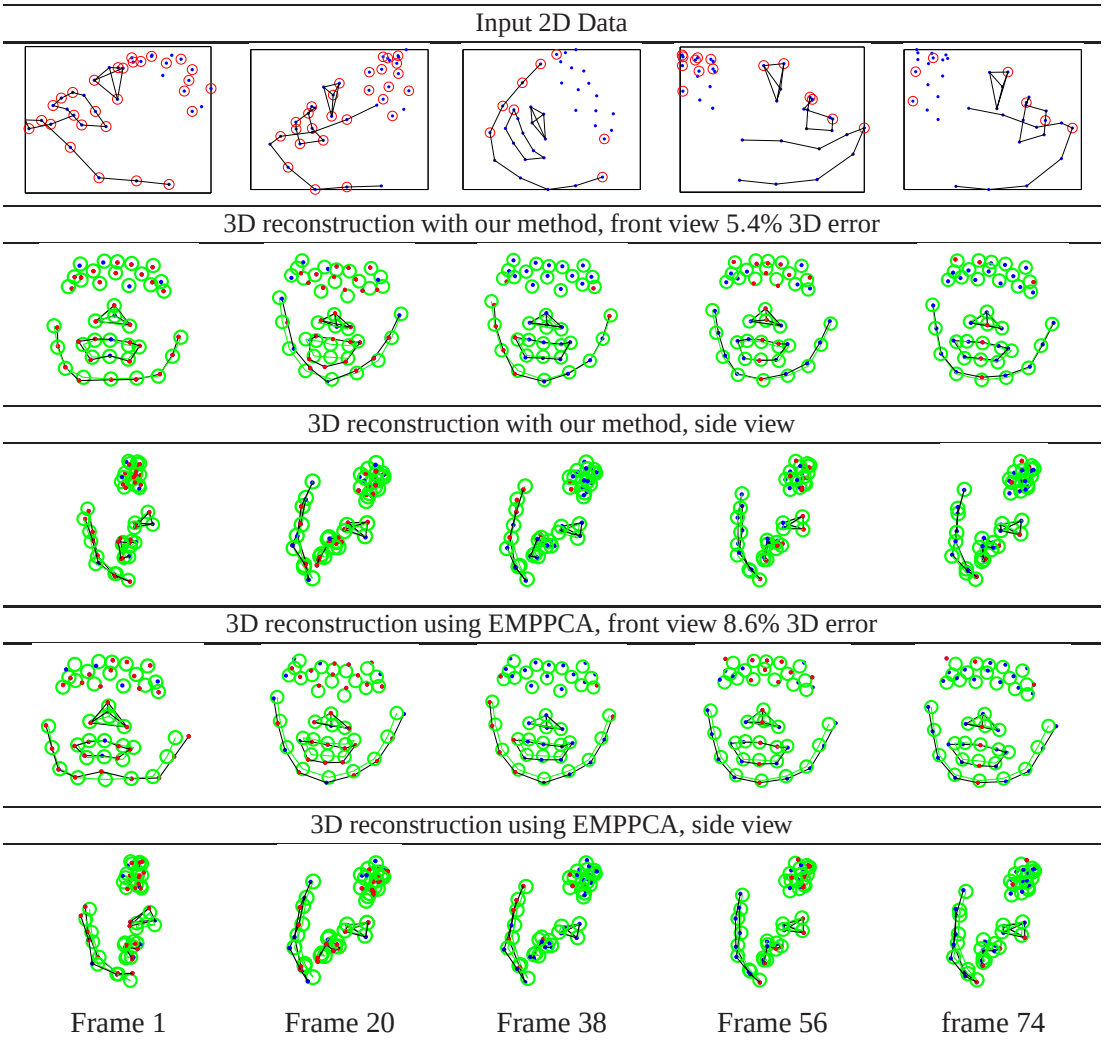


Figure 3.12: 3D reconstruction results obtained for the *Face1* motion capture sequence with the structured missing data mask shown in Figure 3.7. Comparison between Metric Projection (second and third rows) and EMPPCA (fourth and fifth rows). Ground truth 3D data points shown in green, red dots highlight missing data.

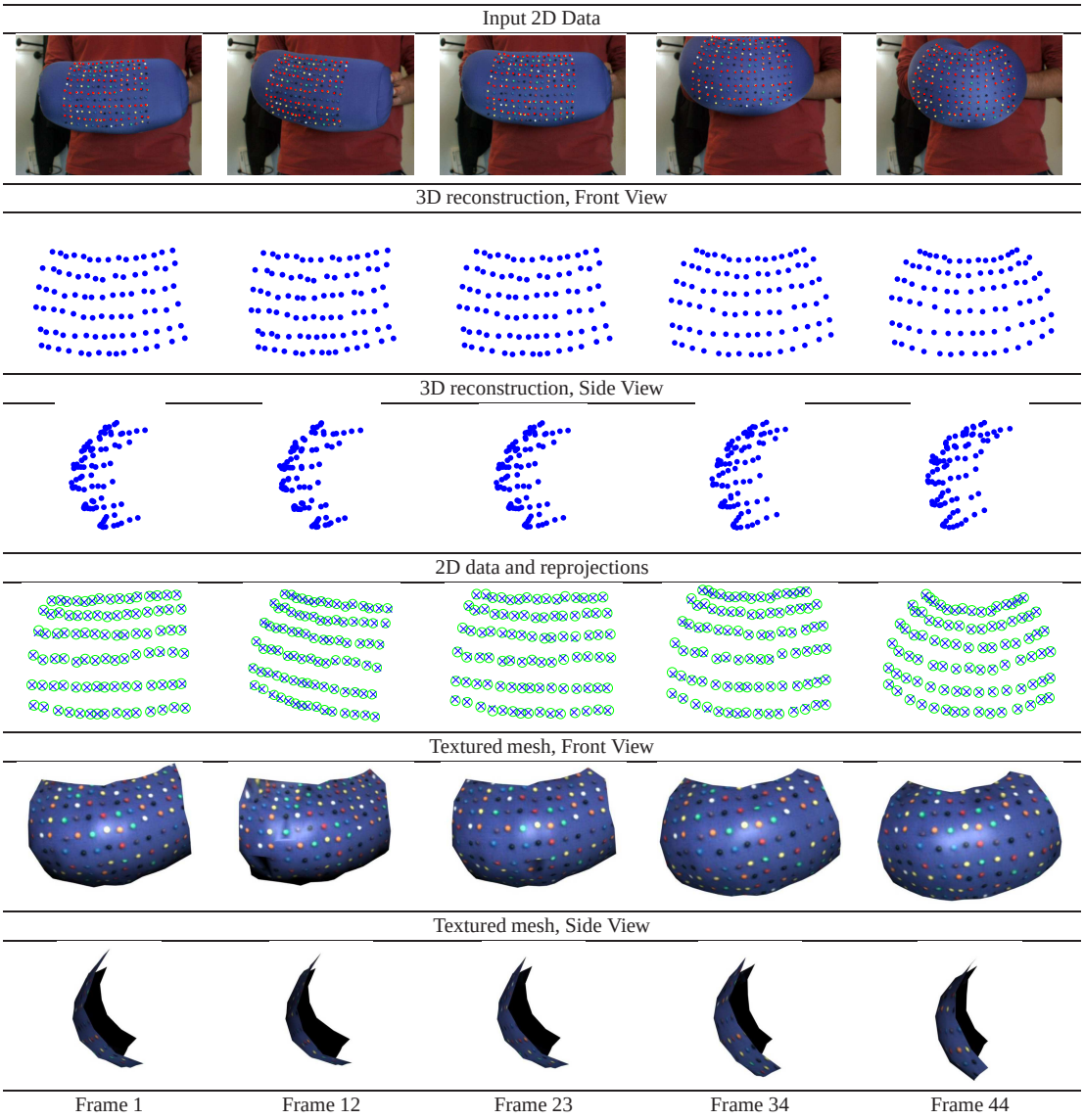


Figure 3.13: 3D reconstruction results for the "cushion" real sequence. We show texture-mapped 3D reconstructions and use them to generate a virtual view of the object in 3D. First row: Input images and tracking data. Second and third rows: 3D reconstruction results with the proposed method. Fourth row: reprojection of reconstructed points (crosses) together with 2D input data (circles). Bottom rows: Texture-mapping rendered view of the 3D reconstruction.

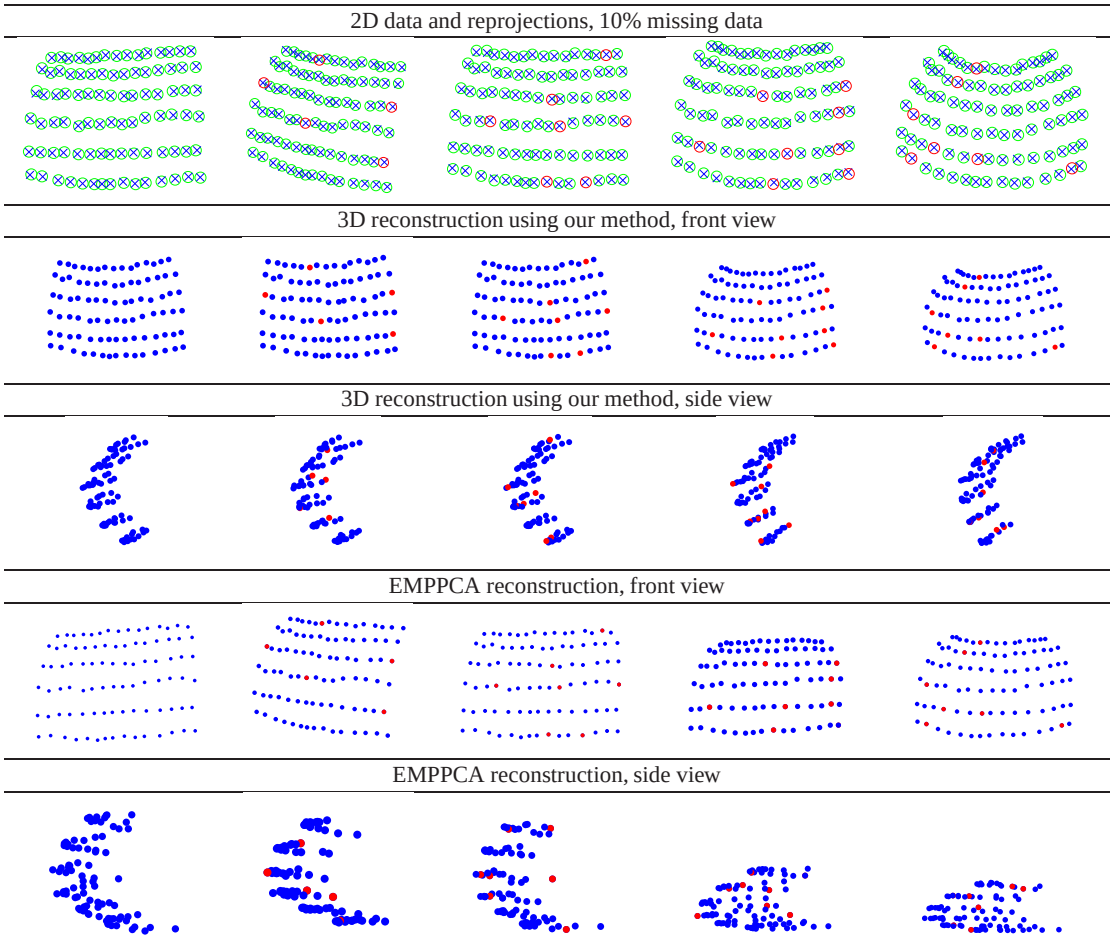


Figure 3.14: Reconstruction results on the "cushion" real sequence with 10% missing data. Points were marked as not visible randomly. First row: Input 2D tracks (green circles) and reprojections calculated with our method (blue crosses). Missing 2D points (not used for reconstruction) are shown as red circles. Second and Third rows: 3D reconstruction with our method. Fourth and Fifth: 3D reconstruction using EMPPCA. note that although the frontal view matches the input data, the reconstruction suffers from bad depth estimation, visible in the side view.

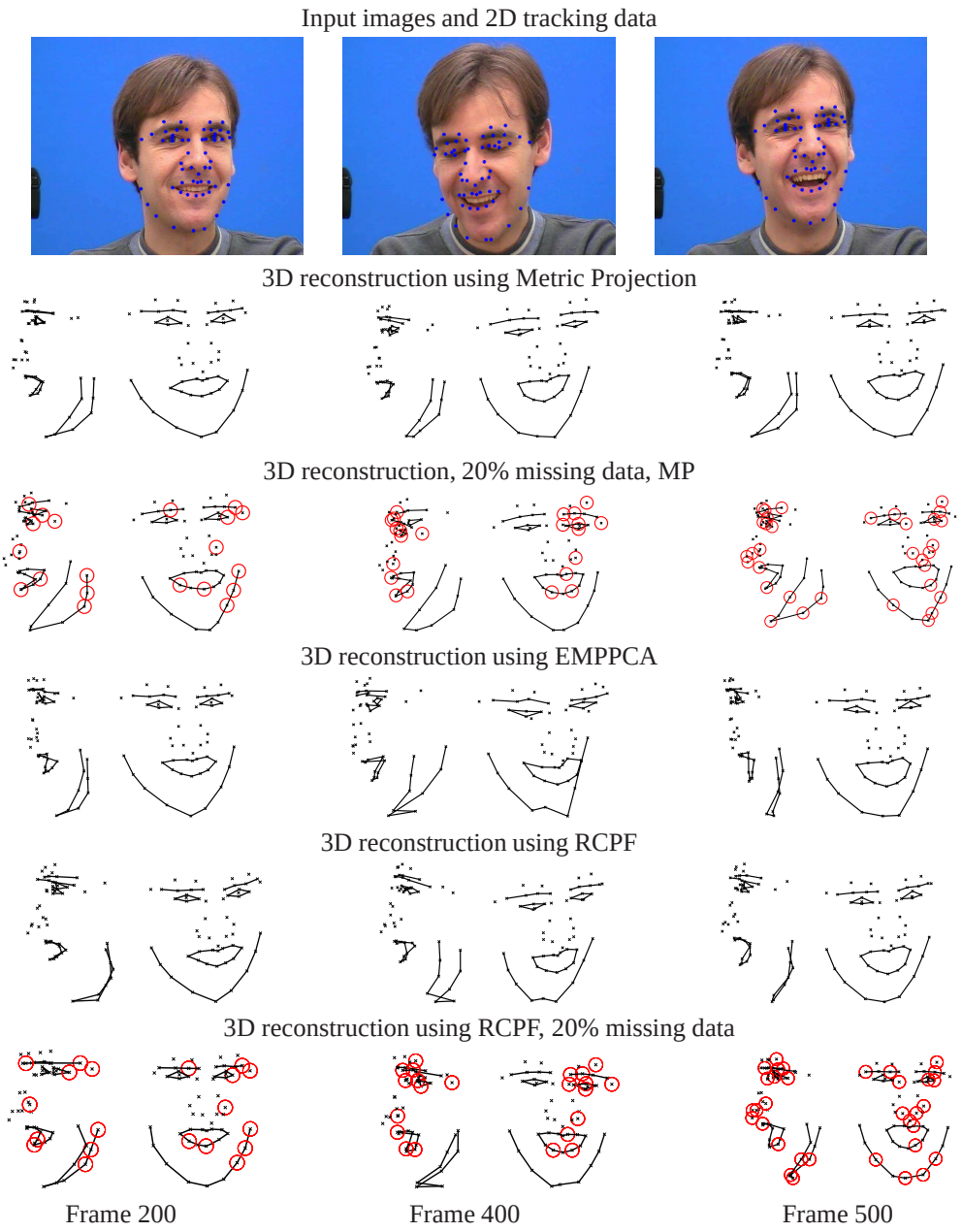


Figure 3.15: First row shows frames 200, 400 and 500 of the Franck sequence. We show front and side views of the 3D reconstructions in the case of full data and 20% missing data in the input tracks (randomly generated) achieved with our MP algorithm (second and third rows) EMPPCA (fourth row) and RCPF (fifth and sixth rows). Note that we do not show the reconstruction obtained for EMPPCA with missing data as it was of very poor quality. Missing points not visible in the corresponding frame are highlighted with a red circle.

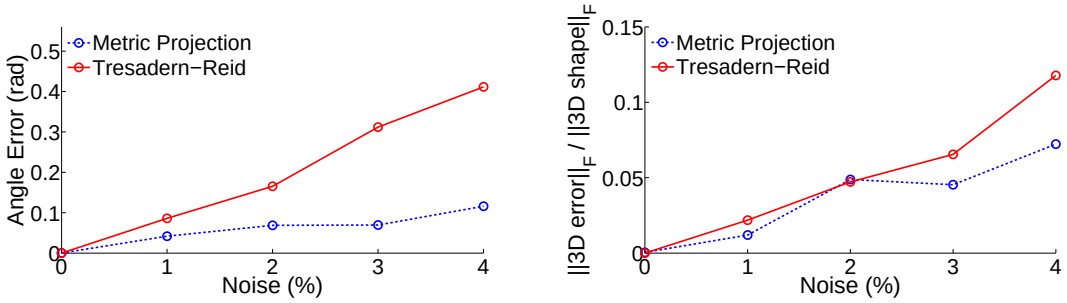


Figure 3.16: Quantitative results on the synthetic articulated sequence. Top: Error on relative rotation angle between the two boxes in the synthetic experiment compared with Tresadern and Reid's linear approach. Bottom: 3D error of recovered structure. In both cases the Metric Projection method results more robust to noise and can recover rotation angles reliably.

3.5.2 Articulated Structure

Synthetic sequence

In the articulated case our synthetic data simulated two 3D boxes coupled by a hinge joint. The 3D ground truth is projected on the input images via orthographic projection. The sequence contained global rotation and translation as well opening and closing of the hinge. Each box contains 231 points, and the sequence is 63 frames long. We tested the algorithm in the case of full data for noise levels ranging from 0% to 4%. Figure 3.16 shows the absolute error in the recovered relative angle between the two boxes (averaged over all frames) and the 3D error of recovered 3D structure. The plots in Figure 3.16 show comparative results between the performance of [113] (TR) and our new approach (MP). Slightly superior results are achieved with our algorithm.

Real Sequence

We tested our algorithm on a sequence of 815 frames of two boxes linked by a hinge joint. The number of tracked points on the upper box was 21 and 47 on the lower box. Figure 3.17 shows two frames of the image sequence showing the tracked points and the recovered joint axis projected onto the images. The 3D reconstruction of the articulated structure together with the common hinge axis is also shown in Figure 3.17. In this case there was no missing data.

Finally we show results using a motion capture sequence of a person kicking a football. The

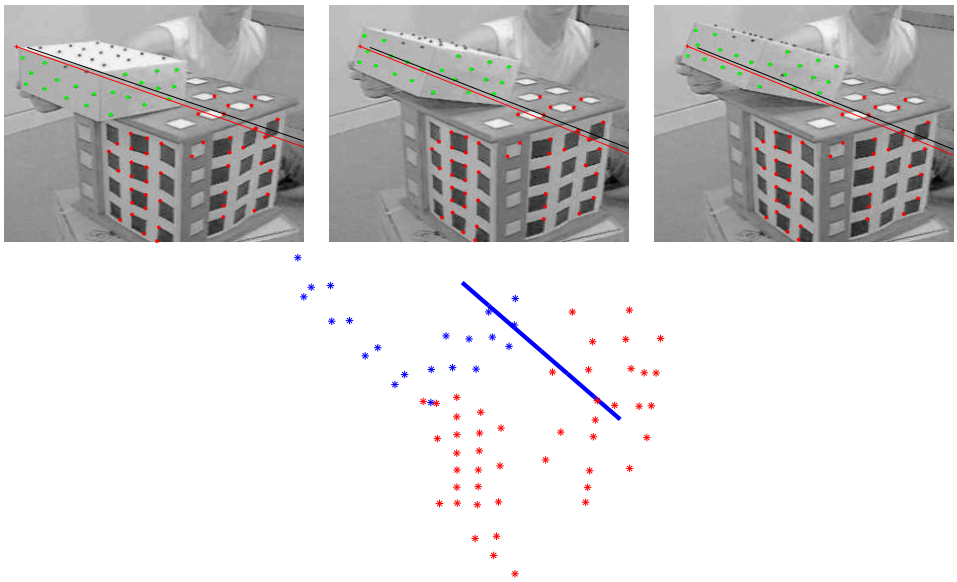


Figure 3.17: Three images from the articulated sequence. The black line represents the hinge location computed with the linear algorithm by Tresadern and Reid, while the red line is the solution given by our method. The last figure shows the final 3D reconstruction and axis obtained using our approach. Images and tracking data kindly provided by Phil Tresadern.

motion capture system tracked 333 markers on the whole body. We selected the tracks on the leg, and projected the 3D coordinates on 2D images via orthographic projection. The viewing direction of the synthetic camera starts at the back of the leg and performs a random rotation around the body, resulting in the image sequence used for reconstruction. Some frames can be seen in Figure 3.19, first row. From the 2D images we can recover the rotation axis of the joint, and the 3D structure of the leg, as shown in Figure 3.19. The reconstructed 3D points and axis have been aligned to the MOCAP data to show the full body pose. Two close-up of the reconstruction and axis are shown. In Figure 3.18 we also show a comparison of the recovered rotation angle between our method and the linear method by Tresadern and Reid [113]. We can see that although this sequence does not have ground truth information on the joint angle in the knee, we recover a smooth movement (purely from the data, without imposing smoothness constraints) while the linear solution obtains similar values with some discontinuities.

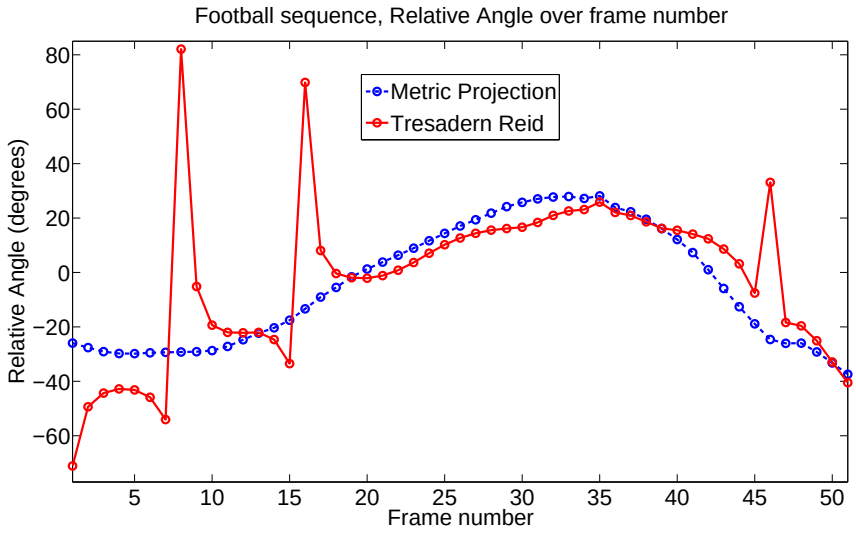


Figure 3.18: Recovered rotation angle between two object: knee joint in the "football" sequence. Although this sequence does not have ground truth information on the joint angle in the knee, we recover a smooth movement (purely from the data, without imposing smoothness constraints) while the linear solution obtains similar values with some discontinuities

3.6 Summary and discussion

We have described a new bilinear alternating approach associated with a globally optimal projection step onto the manifold of metric constraints. At each step of the minimisation we project the motion matrices onto the correct deformable or articulated metric *motion manifolds* respectively. Although the constraints result in non-convex problems we introduced efficient convex relaxations in the form of semi-definite (SDP) or second-order cone (SOCP) programs. These relaxations revealed themselves to be exact in all our numerical experiments.

We have carried out experiments to compare the performance of our new Metric Projection algorithm with competing NRSfM methods. These have revealed that there are two main factors that make our Metric Projection (MP) algorithm more robust to missing data. The first strength is in the projector. It was first observed by Marques and Costeira [70], in the case of rigid SFM, that projecting the rotation matrices onto the Stiefel manifold allowed to cope with high percentages of missing data and degeneracies. Our experimental results show that, in the non-

rigid case, the two algorithms that project the orthographic camera matrices onto the Stiefel manifold: our own MP and the simpler rotation constrained powerfactorization (RCPF) [123] can cope with higher levels of missing data tracks than the two other baseline methods that do not (EMPPCA [112] and Bundle Adjustment [36]). However, MP consistently outperforms RCPF [123] for percentages of missing data above 50%.

This is due to the second strength of our MP algorithm: it simultaneously estimates the unknown entries of the measurement matrix $\mathtt{W}$, given the current estimates of the model parameters, within an iterative outer loop. Differently, RCPF, BA and EMPPCA estimate the model parameters using only the known data. This can have a negative effect on the minimisation when few data are known. We also observed that, when included within our outer iterative loop to deal with missing data, the simple projector used by Wang *et al*. [123] improved its performance significantly for percentages of missing data higher than 50%.

To conclude, imposing the metric constraints on the motion matrices provides reliable results without the need to impose additional smoothness priors on the camera pose or the deformations as most other NRSfM approaches to avoid ambiguous solutions. In the articulated case, we efficiently compute the joints given the non-linear constraints on the motion of the two bodies. In general, even though our methods were designed to solve SfM problems, the *motion manifolds* and the related optimal projections could be used for different tasks such as registration (where the shape $\mathtt{S}$ is known), image point matching and motion segmentation.

The methods described in this chapter, and the experimental results obtained, demonstrated that the manifold constraints are the heart of the non-rigid structure from motion problem. We proposed a unified formulation, to address rigid, deformable, and articulated structures, within the same estimation framework. This idea can be extended, as we are going to describe in the next chapter, the manifold constraints are a powerful tool that can be exploited in a wide variety of problems. The next chapter will deal with a general framework to solve bilinear problems with manifold constraints in computer vision, and in other fields.

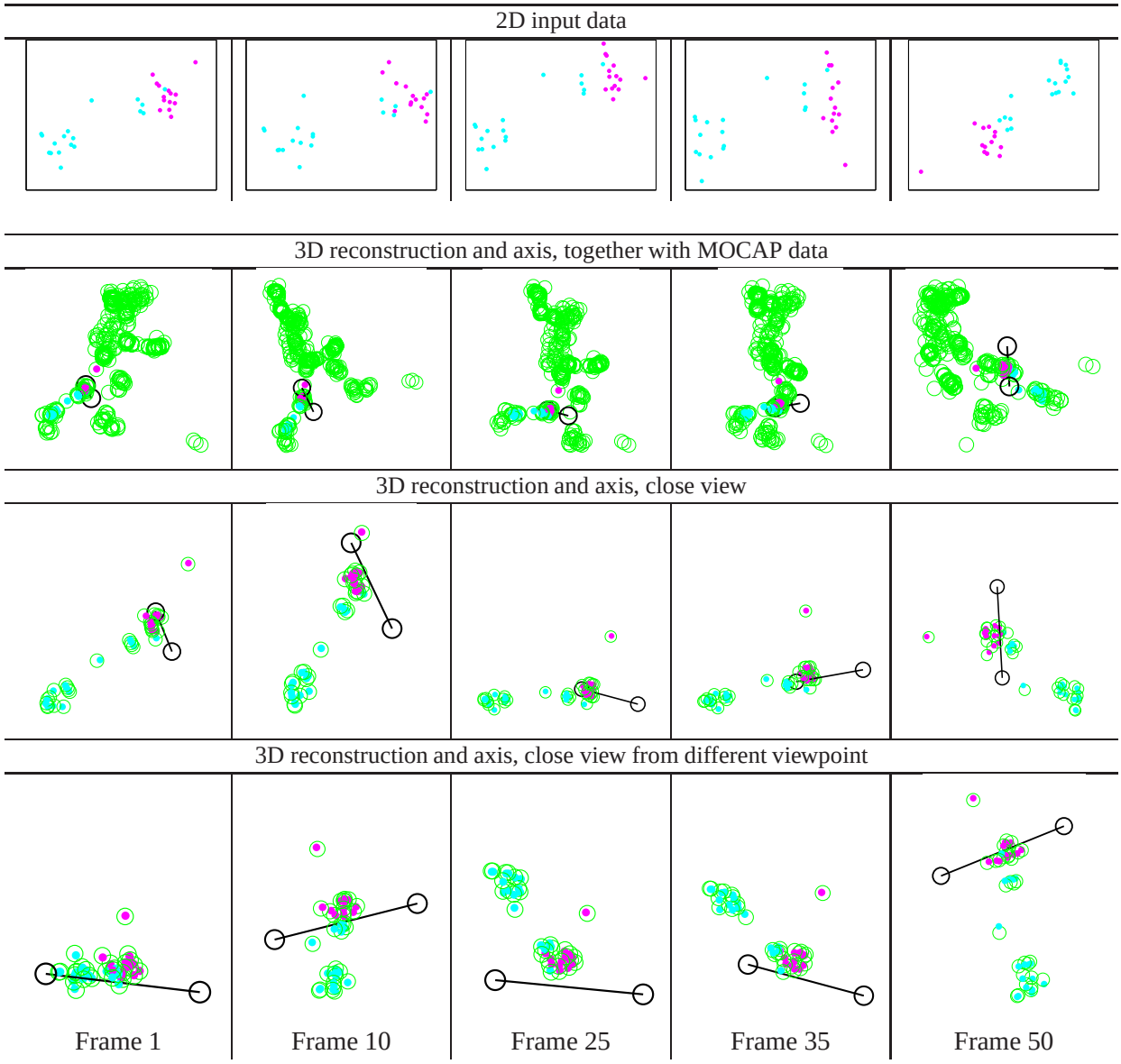


Figure 3.19: Recovery of the knee joint in the "football" sequence. Top row: Input image points. Second row: 3D Reconstruction of the leg (magenta and cyan dots) and axis of rotation shown with the 3D ground truth motion capture sequence (green circles). Third row: Reconstructed 3D points (dots) with ground truth MOCAP data (green circles). Fourth row: 3D reconstruction imaged from a different angle.

Chapter 4

Bilinear modelling via Augmented Lagrange Multipliers (BALM)

Our metric projection work has shown how a unified approach to deformable and articulated object reconstruction via bilinear alternation is possible by considering the projection onto the manifold of acceptable solutions defined by the problems. What follows is a more general solution that deals robustly with missing data and definitely decouples the problem of bilinear estimation from the projection, hence opening the road for a unified framework for the solution of a wide range of computer vision problems.

This chapter presents a unified approach to solve bilinear factorisation problems in the presence of missing data in the measurements. Bilinear problems are common in computer vision. Rigid, articulated and deformable structure from motion all share this formulation. The difference is in the constraints that must be satisfied by one of the factors — the manifold on which the solution lies. Thus, intuitively, it should be possible to construct a unified optimisation framework in which a change of the manifold constraint just implies replacing an inner module of the algorithm (as opposed to an overall redesign of the optimisation method from scratch). The proposed

solution is a constrained optimisation method where one of the factors is constrained to lie on a specific manifold. To achieve this, we introduce an equivalent reformulation of the bilinear factorisation problem. This reformulation decouples the core bilinear aspect from the manifold specificity. We then tackle the resulting constrained optimisation problem with the method of Augmented Lagrange Multipliers (ALM). One advantage of this algorithm is that only a projector onto the manifold constraint is needed. That is the strength and the novelty of this approach: this framework can handle seamlessly different computer vision problems. What will differ in each case is the projector of the solution onto the correct manifold. If the manifold projector exists, the factorisation problem can be formulated using our unified approach. Since in the previous chapter we have proposed projectors for both the deformable and articulated motion manifolds we use them here to solve the non-rigid and articulated SfM problems within this Augmented Lagrange Multipliers (ALM) framework.

4.1 Introduction

Several computer vision problems are naturally formulated as bilinear problems since often observations are influenced by two independent factors where each can be described by a linear model. For example in photometric stereo [10] the shape of the object and the light source direction interact bi-linearly to influence the image intensity. In rigid structure from motion [110] the 3D shape of the object is pre-multiplied by the camera matrix to determine its image coordinates. In structure from sound the time arrival of a sound event depends both on the direction of the sound propagation and the position of the microphones [108]. In facial tracking the problem of separating head pose and facial expression can also be defined as a bilinear problem [9]. In non-rigid structure from motion [15] the 2D coordinates of features arise from a bilinear relation between the camera matrix and the time varying shape. All these are common bilinear problems, where the goal is the simultaneous estimation of two factors.

In our experiments we show that we are able to deal with high percentages of missing data which has the practical implication that our approach can be used on data coming from real, not just

controlled, scenarios. We illustrate our unified approach by applying it to the computer vision problems addressed in this thesis: rigid, articulated and non-rigid structure from motion.

4.2 Related Work

Bilinear models appear frequently in Computer Vision. However, it is in the area of Structure from Motion (SfM) that most of the efforts dedicated to solve this problem have come from. We focus on describing what we believe are the two most important threads of research to solve the problem of low-rank matrix factorisation in the case of missing data.

One line of research that dominates the literature includes approaches that perform alternation of closed-form solutions to solve for the two factors of the matrix. The first of these approaches to solve the problem of missing data was proposed by Wiberg [125]. Since then many different solutions have been put forward. Buchanan and Fitzgibbon [17] provide a comprehensive review of these methods while proposing their own alternative approach. Their Damped Newton algorithm provides faster and more accurate solutions than standard alternation approaches. The common property of all these methods is that they only solve the low-rank matrix factorisation problem without imposing manifold constraints. The constraints are applied afterwards, once the low-rank matrix has been estimated. Crucially, the constraints are not imposed during the minimisation.

On the other hand, a relatively recent set of algorithms have attempted to solve the problem by including explicitly the non-linear constraints given by the specific problem structure in the low-rank minimisation. Marques and Costeira [70] introduced the concept of *motion manifold* in rigid SfM to obtain motion matrices that exactly satisfy the camera constraints. Similarly, Paladini *et al*. [85] propose an alternation algorithm associated with an optimal projector onto the *motion manifold* of non-rigid shapes. The practical implication of their algorithm is that it can deal with very high percentages of missing data. Shaji *et al*. [100] also propose to solve a non-linear optimisation problem directly on the product manifold of the Special Euclidean Group claiming better results than [17] in a rigid real sequence.

However, all these approaches are tailored to specific problems. Therefore, for different manifold constraints an overall redesign of the optimisation method would be needed. The purpose of this work is to present a generic approach that is not problem dependent. In similar spirit, Chandraker and Kriegman [19] have proposed a globally optimal bilinear fitting approach for general Computer Vision problems. The key contribution of their approach is that they can prove convergence to a global minimiser using a branch and bound approach. However, the main drawback is that the scenarios to which their method can be applied are restricted to simple bilinear problems where the number of variables in one of the sets must be very small (for instance just 9 variables in one of their examples). Although their method is very interesting from a theoretical point of view, it only provides practical solutions for problems with a very small number of variables.

This Bilinear factorisation via Augmented Lagrange Multipliers (BALM) is designed to deal with large-scale optimisation problems with the inclusion of non-linear constraints. This is not the first approach to adopt the Augmented Lagrangian Multipliers (ALM) framework in the Computer Vision or related contexts. In perspective 3D reconstruction [69] ALM was used to enforce constraints on the perspective depths. In [63] ALM is successfully employed as a single matrix imputation algorithm which can deal with large scale problems.

4.3 Problem statement

We denote by $\mathtt{Y} \in \mathbb{R}^{n\times m}$ the measurement matrix. In this work, we consider the general case of missing data. We let the finite set $\mathcal{O} := \{(i,j) : \mathtt{Y}_{ij} \text{ is observed}\}$ enumerate the indices of the entries of $\mathtt{Y}$ which are available. The bilinear factorisation problem we address is the following constrained optimisation problem:

$$\begin{aligned} &\text{minimise} && \textstyle\sum_{(i,j)\in\mathcal{O}} \left(\mathtt{Y}_{ij} - s_i^\top m_j\right)^2 \\ &\text{subject to} && \mathtt{M}_i \in \mathcal{M}, \quad i = 1,\ldots,f, \end{aligned} \tag{4.1}$$

where $s_i^\top$ denotes the ith row of the matrix $S \in \mathbb{R}^{n\times r}$ and m_j denotes the jth column of the matrix

$\mathtt{M} = \begin{bmatrix} \mathtt{M}_1 & \cdots & \mathtt{M}_i & \cdots & \mathtt{M}_f \end{bmatrix} \in \mathbb{R}^{r\times m}, \quad \mathtt{M}_i \in \mathbb{R}^{r\times p}$. Note that we are using a dummy variable i twice in (4.1): in the cost function (i.e., $(i,j) \in \mathcal{O}$) and in the constraints (to enumerate the sub-matrices $\mathtt{M}_i$ of $\mathtt{M}$).

The variables in (4.1) are $(\mathtt{S},\mathtt{M})$. In the structure-from-motion problem, f is the number of frames. We consider a generic bilinear problem in which each sub-block of the $\mathtt{M}$ matrix has to satisfy the manifold constraints. In the structure from motion problem, $\mathtt{S}$ will be the 3D structure and $\mathtt{M}$ the camera matrices, in photometric stereo $\mathtt{S}$ would be the lighting parameters and $\mathtt{M}$ the surface normals and albedo.

In words, problem (4.1) consists in finding the best rank r factorisation of $\mathtt{Y}$, given the available entries enumerated by $\mathcal{O}$ and subject to the manifold constraints on $\mathtt{M}$. More precisely, each sub-matrix $\mathtt{M}_i \in \mathbb{R}^{r\times p}$ must belong to the manifold $\mathcal{M} \subset \mathbb{R}^{r\times p}$. Our aim in this work is to construct an algorithm to solve problem (4.1) which takes advantage of the projection onto $\mathcal{M}$. That is, we assume that, for a given $\mathtt{A} \in \mathbb{R}^{r\times p}$, it is known how to solve the projection problem onto $\mathcal{M}$

$$\begin{aligned} &\text{minimise} \quad \|\mathtt{A}-\mathtt{X}\|^2 \ , \\ &\text{subject to} \quad \mathtt{X} \in \mathcal{M} \end{aligned} \tag{4.2}$$

where $\|\mathtt{X}\|$ denotes the Frobenius norm of $\mathtt{X}$. In the rest of the chapter we will denote $p_{\mathcal{M}}(\mathtt{A})$ a solution of (4.2). The role of the projector can be visualised in Figure 4.1.

Problem reformulation. Let us define a new set of variables $z := \{\mathtt{Z}_{ij} : (i,j) \notin \mathcal{O}\}$. Those can be used to represent the non-observed entries of $\mathtt{Y}$. We can introduce these variables in (4.1) and obtain the following equivalent optimisation problem

$$\begin{aligned} &\text{minimise} \quad \|\mathtt{Y}(z) - \mathtt{SM}\|^2 \\ &\text{subject to} \quad \mathtt{M}_i \in \mathcal{M}, \quad i = 1,\ldots,f, \end{aligned} \tag{4.3}$$

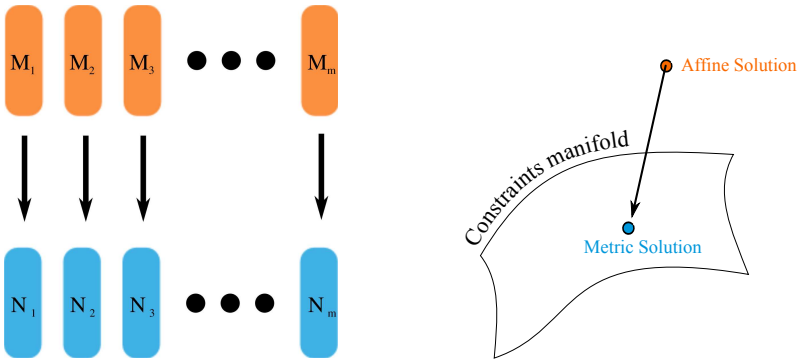


Figure 4.1: A visual representation of the manifold projector. The manifold constraints are assigned to the $\mathtt{N}_i$ variables, which can be computed as the manifold projection of $\mathtt{M}_i$ onto the manifold of the problem constraints.

where the (i,j) entry of the matrix $\mathtt{Y}(z)$ is defined as

$$(\mathtt{Y}(z))_{ij} := \begin{cases} \mathtt{Y}_{ij} & \text{, if } (i,j) \in \mathcal{O} \\ \mathtt{Z}_{ij} & \text{, if } (i,j) \notin \mathcal{O} \end{cases}.$$

In words, $\mathtt{Y}(z)$ is the input data $\mathtt{Y}$ with a filling of the missing entries given by z. Note that the variables to optimise in (4.3) are $(z,\mathtt{S},\mathtt{M})$. Problem (4.3) is equivalent to (4.1) because once we fix $(\mathtt{S},\mathtt{M})$ in (4.3) and minimise over z we fall back into (4.1). Finally, we add a new set of variable to deal with the manifold constraints. We clone $\mathtt{M}$ into a new variable $\mathtt{N} = \begin{bmatrix} \mathtt{N}_1 & \cdots & \mathtt{N}_i & \cdots & \mathtt{N}_f \end{bmatrix} \in \mathbb{R}^{r\times m}$, $\mathtt{N}_i \in \mathbb{R}^{r\times p}$, and transfer the manifold constraint to the latter. By doing so, we roughly separate the bilinear estimation from the manifold constraints. Thus our reformulation becomes:

$$\begin{aligned} \text{minimise} \quad & \|\mathtt{Y}(z) - \mathtt{SM}\|^2 \\ \text{subject to} \quad & \mathtt{M}_i = \mathtt{N}_i, \quad i = 1,\ldots,f \\ & \mathtt{N}_i \in \mathcal{M}, \quad i = 1,\ldots,f. \end{aligned} \tag{4.4}$$

With this formulation the variables to estimate in (4.4) are $(z,\mathtt{S},\mathtt{M},\mathtt{N})$.

4.4 The BALM algorithm

The main difficulty in the constrained optimisation problem (4.4) are the equality constrains $\mathtt{M}_i = \mathtt{N}_i$. We propose to handle them through an augmented Lagrangian approach, see [54, 11] for details on this optimisation technique. In our context, the augmented Lagrangian corresponding to (4.4) is given by

$$\begin{aligned} L_\sigma(z,\mathtt{S},\mathtt{M},\mathtt{N};\mathtt{R}) &= \|\mathtt{Y}(z)-\mathtt{S}\mathtt{M}\|^2 - \\ &\sum_{i=1}^{f} \mathtt{tr}\left(\mathtt{R}_i^\top(\mathtt{M}_i-\mathtt{N}_i)\right) + \frac{\sigma}{2}\sum_{i=1}^{f}\|\mathtt{M}_i-\mathtt{N}_i\|^2 . \end{aligned} \tag{4.5}$$

where $\sigma > 0$ is the weight of the penalty term and matrix $\mathtt{R} = \begin{bmatrix} \mathtt{R}_1 & \dots & \mathtt{R}_f \end{bmatrix}$ contains the Lagrange multipliers $\mathtt{R}_i$, $i = 1,\dots,f$. The optimisation problem (4.4) can then be tackled by our Bilinear factorisation via Augmented Lagrange Multipliers (BALM) algorithm detailed in Algorithm 3.

Clearly, solving the inner problem (4.6) at each iteration of the BALM method is the main computational step. Note that in (4.6) the optimisation variable is $(z,\mathtt{S},\mathtt{M},\mathtt{N})$ ($\sigma^{(k)}$ and $\mathtt{R}^{(k)}$ are constants). To tackle (4.6) we propose an iterative Gauss-Seidel scheme which is described in Algorithm 4. We now show that each of the sub-problems (4.7), (4.8) and (4.9) inside the Gauss-Seidel scheme are easily solvable.

4.4.1 Solving for the manifold constraints

Problem (4.7) requires a minimisation over $\mathtt{N}_i \in \mathcal{M}$, $i = 1,\dots,f$, the remaining variables being constant. Thus, by using (4.5) and dropping the constant terms, problem (4.7) becomes equivalent to

$$\begin{aligned} \mathtt{N}^{[l+1]} = \quad & \text{argmin} \quad \textstyle\sum_{i=1}^{f}\left\|\mathtt{N}_i - \left(\mathtt{M}_i^{[l]} - \frac{1}{\sigma^{(k)}}\mathtt{R}_i^{(k)}\right)\right\|^2 . \\ & \text{subject to} \quad \mathtt{N}_i \in \mathcal{M}, \quad i = 1,\dots,f \end{aligned} \tag{4.10}$$

Algorithm 3 Bilinear factorisation via Augmented Lagrange Multipliers (BALM)

set $k = 0$ and $\varepsilon_{\text{best}} = +\infty$
initialise $\sigma^{(0)}$, $\mathtt{R}^{(0)}$, $\gamma > 1$ and $0 < \eta < 1$
initialise $z^{(0)}$, $\mathtt{S}^{(0)}$ and $\mathtt{M}^{(0)}$
repeat
    solve

$$\begin{aligned}\left(z^{(k+1)}, \mathtt{S}^{(k+1)}, \mathtt{M}^{(k+1)}, \mathtt{N}^{(k+1)}\right) &= \\ = \quad \underset{}{\text{argmin}} \quad & L_{\sigma^{(k)}}(z, \mathtt{S}, \mathtt{M}, \mathtt{N}; \mathtt{R}^{(k)}) \\ \text{subject to} \quad & \mathtt{N}_i \in \mathcal{M}, \quad i = 1, \ldots, f,\end{aligned} \tag{4.6}$$

    using the iterative Gauss-Seidel scheme described in **Algorithm 4**
    compute $\varepsilon = \left\|\mathtt{M}^{(k+1)} - \mathtt{N}^{(k+1)}\right\|^2$
    if $\varepsilon < \eta\, \varepsilon_{\text{best}}$
        $\mathtt{R}^{(k+1)} = \mathtt{R}^{(k)} - \sigma^{(k)}\left(\mathtt{M}^{(k+1)} - \mathtt{N}^{(k+1)}\right)$
        $\sigma^{(k+1)} = \sigma^{(k)}$
        $\varepsilon_{\text{best}} = \varepsilon$
    else
        $\mathtt{R}^{(k+1)} = \mathtt{R}^{(k)}$
        $\sigma^{(k+1)} = \gamma\sigma^{(k)}$
    endif
    update $k \leftarrow k + 1$
until some stopping criterion

That is, problem (4.7) decouples into f projections onto the manifold of constraints $\mathcal{M}$. More precisely, if we partition

$$\mathtt{N}^{[l+1]} = \left[\mathtt{N}_1^{[l+1]} \quad \cdots \quad \mathtt{N}_i^{[l+1]} \quad \cdots \quad \mathtt{N}_f^{[l+1]}\right] \in \mathbb{R}^{r\times m},$$

with $\mathtt{N}_i^{[l+1]} \in \mathbb{R}^{r\times p}$. The solution of (4.7) is given by

$$\mathtt{N}_i^{[l+1]} = p_{\mathcal{M}}\left(\mathtt{M}_i^{[l]} - \frac{1}{\sigma^{(k)}}\mathtt{R}_i^{(k)}\right), \quad i = 1, \ldots, f. \tag{4.11}$$

We recall that $p_{\mathcal{M}}$ stands for the projector onto $\mathcal{M}$, see (4.2), which we assume is available. This is the only part of the algorithm where the constraint manifold $\mathcal{M}$ plays a role. Thus, replacing $\mathcal{M}$ amounts to replace the projector $p_{\mathcal{M}}$. This is the modularity which is key to the

Algorithm 4 Iterative Gauss Seidel scheme to solve for (4.6)

1: set $l = 0$ and choose $L_{\max}$
2: set $z^{[0]} = z^{(k)}$, $\mathtt{S}^{[0]} = \mathtt{S}^{(k)}$ and $\mathtt{M}^{[0]} = \mathtt{M}^{(k)}$
3: **repeat**
4: solve

$$\begin{aligned} &\mathtt{N}^{[l+1]} = \\ &= \begin{array}{ll} \text{argmin} & L_{\sigma^{(k)}}\left(z^{[l]}, \mathtt{S}^{[l]}, \mathtt{M}^{[l]}, \mathtt{N}; \mathtt{R}^{(k)}\right) \\ \text{subject to} & \mathtt{N}_i \in \mathcal{M}, \quad i = 1, \ldots, f, \end{array} \end{aligned} \tag{4.7}$$

5: solve

$$\begin{aligned} &\left(\mathtt{S}^{[l+1]}, \mathtt{M}^{[l+1]}\right) = \\ &= \text{argmin} \quad L_{\sigma^{(k)}}\left(z^{[l]}, \mathtt{S}, \mathtt{M}, \mathtt{N}^{[l+1]}; \mathtt{R}^{(k)}\right) \end{aligned} \tag{4.8}$$

6: solve

$$\begin{aligned} &z^{[l+1]} = \\ &= \text{argmin} \quad L_{\sigma^{(k)}}\left(z, \mathtt{S}^{[l+1]}, \mathtt{M}^{[l+1]}, \mathtt{N}^{[l+1]}; \mathtt{R}^{(k)}\right) \end{aligned} \tag{4.9}$$

7: update $l \leftarrow l + 1$
8: **until** $l = L_{\max}$
9: set $\mathtt{S}^{(k+1)} = \mathtt{S}^{[L_{\max}]}$, $\mathtt{M}^{(k+1)} = \mathtt{M}^{[L_{\max}]}$ and $\mathtt{N}^{(k+1)} = \mathtt{N}^{[L_{\max}]}$

application of this method to many different bilinear problems.

4.4.2 Solving for the bilinear factorisation

Solving (4.8) corresponds to solving

$$\text{minimise} \quad \left\|\mathtt{Y}\left(z^{[l]}\right) - \mathtt{SM}\right\|^2 + \frac{\sigma^{(k)}}{2} \sum_{i=1}^{f} \left\|\mathtt{M}_i - \left(\mathtt{N}_i^{[l+1]} + \frac{1}{\sigma}\mathtt{R}_i^{(k)}\right)\right\|^2.$$

The solution to this factorisation problem can be found solving 2 least-squares problems, first over $\mathtt{M}$ (fixed $\mathtt{S}$) and then over $\mathtt{S}$ (fixed $\mathtt{M}$). An alternative efficient solution to (4.8) was proposed in [37] based on re-parametrisation the $\mathtt{M}$ matrix as the product of an invertible and a Stiefel matrix. The solution is then obtained via eigenvalue decomposition.

4.4.3 Solving for the missing data

After solving for $\mathtt{N}^{[l+1]}$ and $\left(\mathtt{S}^{[l+1]}, \mathtt{M}^{[l+1]}\right)$, problem (4.9) updates the missing data. The solution of (4.9) is trivial: we just have to take $Z_{ij}^{[l+1]}$ as the (i,j)th entry of $\mathtt{S}^{[l+1]}\mathtt{M}^{[l+1]}$ for all $(i,j) \notin \mathcal{O}$.

4.4.4 Initialisation

Regarding the initialisation of the BALM algorithm, we used $\sigma^{(0)} = 1$, $\mathtt{R}^{(0)} = \mathtt{0}$, $\gamma = 5$ and $\eta = 1/2$ in all our computer experiments. With respect to $z^{(0)}$, $\mathtt{S}^{(0)}$ and $\mathtt{M}^{(0)}$, we feel that there is no universally good method, that is, the structure of $\mathcal{M}$ must taken into account. We discuss the initialisation $\left(z^{(0)}, \mathtt{S}^{(0)}, \mathtt{M}^{(0)}\right)$ for non-rigid and articulated SfM in the experimental section of this chapter.

Algorithm convergence. At best, the BALM algorithm can produce a local minimiser for (4.1). That is, we do not claim that BALM (algorithm 3) converges to a global minimiser. In fact, even the non-linear Gauss-Seidel technique (algorithm 4) is not guaranteed to globally solve (4.6). This is the common situation when dealing with non-convex problems. See [23] for some convergence results on augmented Lagrangian methods.

We have developed the generic BALM algorithm to solve a variety of bilinear computer vision problems. What is required is the knowledge of the manifold constraints that a solution must satisfy, and the availability of a projector onto the manifold. In the previous chapter we derived projector onto the non-rigid and articulated motion manifolds. Therefore we will now demonstrate the BALM algorithms on these specific bilinear problems.

4.5 Example 1: BALM for Rigid and Non-Rigid SfM

We have seen how the non-rigid structure from motion problem was formulated as a matrix factorisation problem by Bregler *et al.* [15] in the case of an orthographic camera. The main assumption is that the 3D shape at any frame can be represented as a linear combination of a set of K fixed basis shapes. Thus the 3D shape at a generic frame i will be given by the linear

combination $\mathtt{S}_i = \sum_{d=1}^{K} l_{id}\mathtt{B}_d$. For the rest of this chapter we will use the same factorisation formalism, but we will solve for the transpose problem (e.g. $\mathtt{W} = \mathtt{MS}$ becomes $\mathtt{W}^\top = \mathtt{S}^\top\mathtt{M}^\top$), such that the problem becomes immediately of the same form as problem (4.1). By referring the image coordinates to their centroid, the projection of the shape at frame i can be expressed as

$$\begin{aligned}\mathtt{Y}_i = \begin{bmatrix} u_{i1} & v_{i1} \\ \vdots & \vdots \\ u_{in} & v_{in} \end{bmatrix} = \left(\sum_{d=1}^{K} l_{id}\mathtt{B}_d\right)\mathtt{Q}_i = \\ = \begin{bmatrix} \mathtt{B}_1 & \dots & \mathtt{B}_K \end{bmatrix}(\mathbf{l}_i \otimes \mathtt{Q}_i) = \mathtt{SM}_i\end{aligned} \tag{4.12}$$

where $\mathtt{Y}_i$ is the $n \times 2$ measurement matrix that contains the 2D coordinates of n image points in frame i, $\mathtt{B}_d$ are the basis shapes of size $n \times 3$, l_{id} are the time varying shape coefficients and $\mathtt{Q}_i$ is the projection matrix for frame i. In the case of orthographic projection, $\mathtt{Q}_i$ is a 3×2 matrix that encodes the first two columns of a rotation matrix (therefore it is a Stiefel matrix). Note that we are defining $\mathtt{M}_i := \mathbf{l}_i \otimes \mathtt{Q}_i$, where $\otimes$ denotes the Kronecker product. Rigid SfM can be instantiated with this framework by imposing $K = 1$, a single basis shape.

By concatenating all the measurements for all the frames into a single matrix we have

$$\begin{aligned}\mathtt{Y} = \begin{bmatrix} \mathtt{B}_1 & \dots & \mathtt{B}_K \end{bmatrix}\begin{bmatrix} \mathbf{l}_1 \otimes \mathtt{Q}_1 & \dots & \mathbf{l}_f \otimes \mathtt{Q}_f \end{bmatrix} = \\ = \mathtt{S}\begin{bmatrix} \mathtt{M}_1 & \dots & \mathtt{M}_f \end{bmatrix} = \mathtt{SM}.\end{aligned} \tag{4.13}$$

Now, we have expressed the measurement matrix as a bilinear interaction between the shape matrix $\mathtt{S}$ of size $n \times 3K$ and the motion matrix $\mathtt{M}$ of size $3K \times 2f$. This form fits exactly the optimisation problem as presented in Eq. (4.1). Therefore, in the NRSFM case, the manifold constraint corresponds to

$$\mathcal{M} = \left\{\mathbf{l} \otimes \mathtt{Q} : \mathbf{l} \in \mathbb{R}^K, \mathtt{Q} \in \mathbb{R}^{3\times 2}, \mathtt{Q}^\top\mathtt{Q} = \mathtt{I}_2\right\}, \tag{4.14}$$

or in other words, the two rows of the rotation matrix $\mathtt{Q}^\top$ must be orthonormal (i.e. it is a Stiefel matrix). To apply our BALM algorithm, a projector onto the non-rigid motion manifold $\mathcal{M}$ is required.

In section 3.3.1 in the previous chapter we derived an exact globally optimal projector onto the non-rigid motion manifold. Del Bue *et al*. [37] recently provided an alternative *approximate* projector onto $\mathcal{M}$ which still provides accurate estimates while being considerably faster.

4.6 Example 2: BALM for Articulated SfM

The problem formulation for a factorisation approach to articulated shape and motion recovery was discussed in the previous chapter, Section 3.2.3. We use the same formulation here for convenience (refer to Section 3.2.3 for details). The measurement matrix of the (segmented) object tracks can be written as the product of a common motion matrix and shape matrix:

$$\mathtt{W} = \left[\begin{array}{c|c} \mathtt{W}_i^{(1)} & \mathtt{W}_i^{(2)} \end{array} \right] = \mathtt{M}_i \mathtt{S} \tag{4.15}$$

The shape of two objects will be encoded in $\mathtt{S}$ and the motion matrix has the form:

$$\mathtt{M}_i = \left[\begin{array}{ccc} \mathbf{u}_i & \mathtt{A}_i & \mathtt{B}_i \end{array} \right] \tag{4.16}$$

for each frame i.

The manifold of acceptable solutions in this problem is defined by the constraints:

$$\begin{aligned} [\mathbf{u}_i \; \mathtt{A}_i] \left[\begin{array}{c} \mathbf{u}_i^\top \\ \mathtt{A}_i^\top \end{array} \right] &= \mathtt{I}_{2\times 2} \\ [\mathbf{u}_i \; \mathtt{B}_i] \left[\begin{array}{c} \mathbf{u}_i^\top \\ \mathtt{B}_i^\top \end{array} \right] &= \mathtt{I}_{2\times 2} \end{aligned} \tag{4.17}$$

4.6.1 Articulated manifold projector

The BALM algorithm is suited to estimate articulated structure from motion of two objects coupled by a hinge joint, as we have seen that it is a bilinear factorisation problem. The projector we used has been already defined in the previous chapter in section 3.3.4. Appendix B shows a convex relaxation to solve for equation 4.18.

$$\min_{\mathbf{u},\mathtt{A},\mathtt{B}} J(\mathbf{u},\mathtt{A},\mathtt{B}) = \|\mathbf{u}-\mathbf{x}\|^2 + \|\mathtt{A}-\mathtt{Y}\|_F^2 + \|\mathtt{B}-\mathtt{Z}\|_F^2 , \tag{4.18}$$

4.7 Experiments

To evaluate our unified algorithm we carry out experiments on the example problems proposed above with both synthetic and real data[1]. The aim of our tests is twofold: to show that the performance of BALM is comparable to the best specialised algorithms and to assess its convergence. In the NRSfM problem we will also assess the resilience of our approach to very high levels of missing data.

4.7.1 Synthetic experiments: NRSfM

First we evaluate the performance of our bilinear algorithm when applied to the NRSfM problem. We consider two different sets of synthetic experiments. The first set of tests is designed to verify the resilience of the algorithm to increasing ratios of missing data. We used a 3D motion capture sequence of a face. The sequence was captured using a VICON system tracking a subject wearing 37 markers on the face to provide 3D ground truth for the evaluation. The 3D points were then projected synthetically onto an image sequence 74 frames long using an orthographic camera model. To test the performance of our algorithm we computed the 3D reconstruction error, defined as the Frobenius norm of the difference between the recovered 3D shape $\mathtt{S}$ and the ground truth 3D shape $\mathtt{S}_{GT}$. The relative 3D error is then computed as: $||\mathtt{S}-\mathtt{S}_{GT}||/||\mathtt{S}_{GT}||$. We

[1]The code for the BALM method and the manifold projectors is available at: http://www.isr.ist.utl.pt/~adb/the-balm/.

subtract the centroid of each shape and align them with Procrustes analysis. We evaluated the performance of the algorithm with respect to noise in the image measurements of up to 6% and up to 90% missing data in a combined test. Zero mean additive Gaussian noise was applied with standard deviation $\sigma = \mathrm{n} \times \mathrm{s}/100$ where n is the noise percentage and s is defined as $\max(Y)$ in pixels. In all experiments the number of basis shapes was fixed to $k = 5$. The results for each level of noise were averaged over 100 trials.

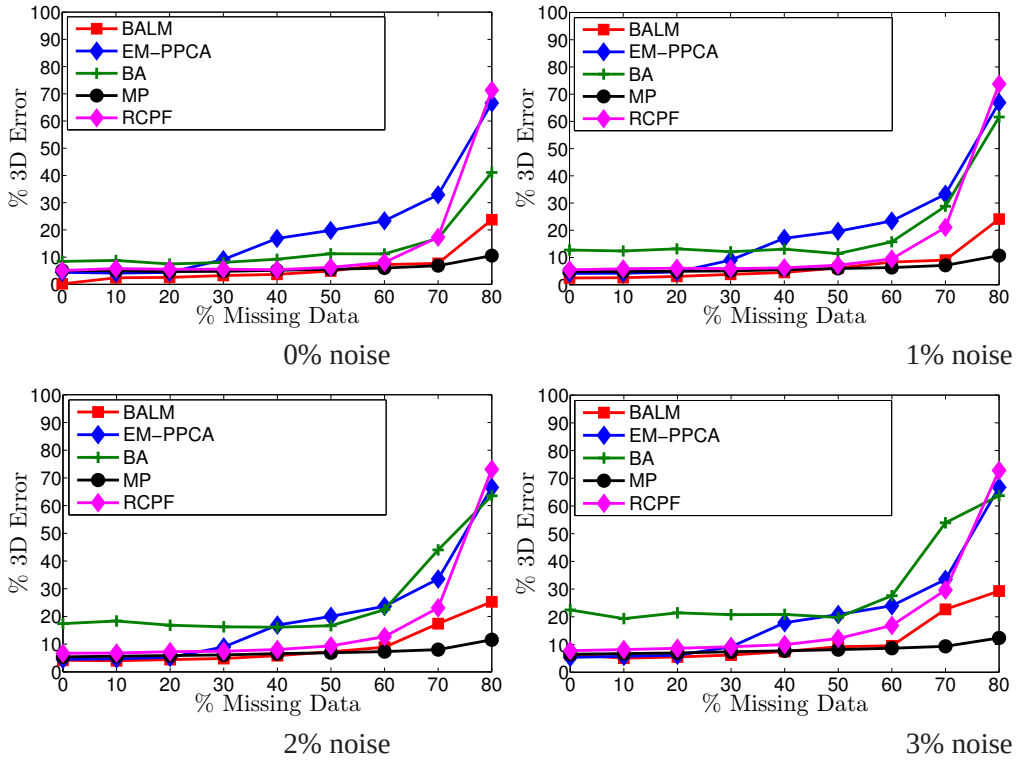


Figure 4.2: Synthetic experiment results showing comparison with several NRSfM methods with different ratios of missing data and noise.

In Figure 4.2 we compare the results of the proposed BALM algorithm with Torresani *et al.*'s algorithm [111] (EM-PPCA), Bundle Adjustment [36], the method of metric projection described in the previous chapter (MP) [85] and the trilinear approach of Wang *et al.* [123] (RCPF) for different levels of noise. In practical terms, note that a reconstruction error above 20% is too high to be of any use in most applications. Regarding the overall results, while in the case of

full data the performance of all algorithms is comparable, BALM and MP outperform the rest of the algorithms in the case of missing data. Notice that RCPF is closer to both BALM and MP since it also includes a projection step of the rotation matrices onto the correct manifold. However, since the projection step is only approximate, this algorithm breaks down for lower levels of missing data. On the other hand, BA and EMPPCA deteriorate for levels of missing data above 30%. Also notice that the algorithms that perform metric projections (BALM, MP and RCPF) are less affected by increasing levels of noise than others. BALM closely follows the performance of the best performing algorithm (MP) which is specific for NRSfM. A noticeable decrease in performance for BALM occurs at 80% missing data (70% for the higher percentages of noise). Regarding run-time, a single manifold projection takes approximately 1.8 msec for each frame with $d = 5$ basis shapes.

Regarding the initialisation of the ALM algorithm in the case of NRSfM, the missing data tracks are first filled in using [70] which enforces metric constraints on the motion matrices. The camera matrices are initialised assuming rigid motion. Torresani *et al.*'s initialisation [111] is then used to estimate the configuration weights and the basis shapes given the residual of the first rigid solution.

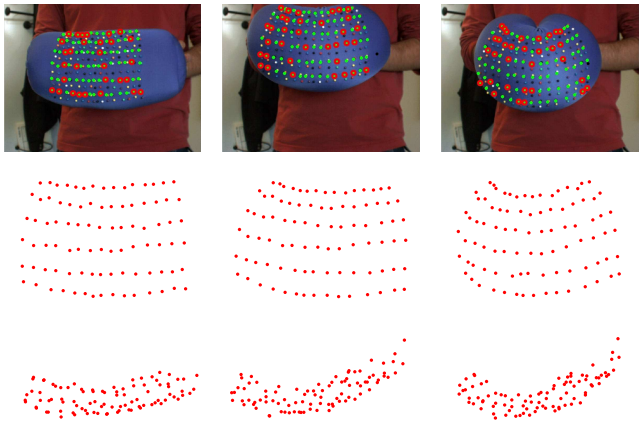


Figure 4.3: Cushion sequence with 40% missing data. First row shows four image samples with missing points highlighted with a red circle. Second and third rows show frontal and side views of the 3D reconstruction using BALM.

4.7.2 Real data: NRSfM

We tested our NRSfM method on a real sequence of a cushion being bent. 90 points were tracked manually for the whole 50-frame long sequence. We simulated a missing data ratio of 40% by eliminating data points randomly. Figure 4.3 shows 4 selected frames and their respective 3D reconstructions (frontal and top view). The bending is clearly observable in the 3D shape plots where BALM shows robustness given the high percentage of missing data.

We have also tested BALM for NRSfM in the face modelling domain on the Franck sequence[2]. The face points were tracked with Active Appearance Models giving 56 points in 700 frames and ratio of 30% missing data was simulated synthetically. The first row of Figure 4.4 shows a sample of the sequence and the bottom row shows the corresponding reconstructions. The resulting 3D shape and deformations describe the shape well, even in occluded areas (e.g. lips).

4.7.3 Real data: Rigid SfM

We have tested our BALM algorithm also in the case of rigid scenes. Figure 4.6 shows results for the dinosaur sequence[3]. Some frames of the sequence are shown in Figure 4.5. Because this sequence contains self-occlusions, there is 76% missing data in the 2D feature tracks. We compare the reconstruction with the methods proposed by Marques and Costeira [70] and Buchanan and Fitzgibbon [17]. Qualitatively our 3D reconstruction recovered the correct shape. Reprojection error results confirm that BALM performs closely to the best performing methods for rigid structure. The overall 2D rms error with BALM was 1.3039[4] which is a slight improvement over the error reported by Marques and Costeira (1.3705) but higher than the error of the Damped-Newton approach by Buchanan and Fitzgibbon (1.0847). Note however that the 2D rms error alone does not provide enough information on the quality of the reconstruction, which can only be compared qualitatively in this real sequence with no ground truth data.

[2]The image sequence is freely available at: www-prima.inrialpes.fr/FGnet/data/01-TalkingFace/talking_face.html

[3]available from `http://www.robots.ox.ac.uk/~vgg/data/data-mview.html`

[4]Notice that in this real sequence the missing data is not simulated, 2D error is calculated only from the known entries.

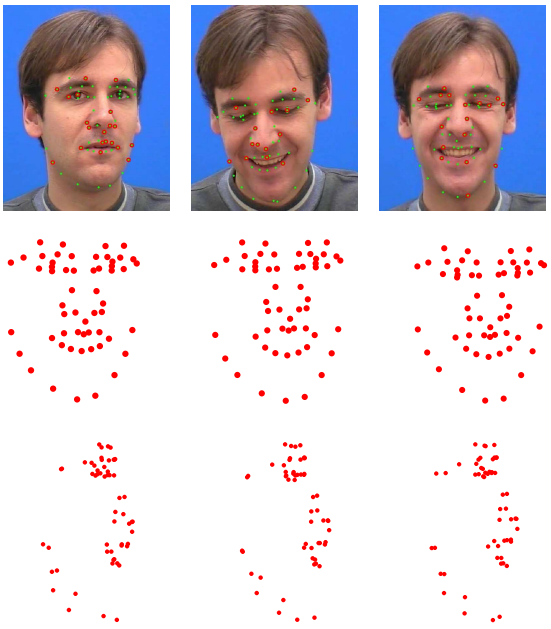


Figure 4.4: The *Franck* sequence (first row) used for our real experiment. Tracked points are in green while red circles show the missing entries. The second row shows the 3D reconstruction of a frontal view with 30% missing data in the input tracks. The third row shows a side view of the 3D shape in order to evaluate the estimated depth.

We have also attempted the reconstruction of the *casa da musica* sequence [70] with 60% missing data where the images were obtained from *Google images* and thus generally shot far apart and with unknown cameras (no temporal consistency of camera views). The sequence also contains degenerate configurations since only planar surfaces are seen from each camera view. The 3D reconstruction in Figure 4.7 shows that most of the planar surfaces are correctly reconstructed and they provide a credible 3D reconstruction.

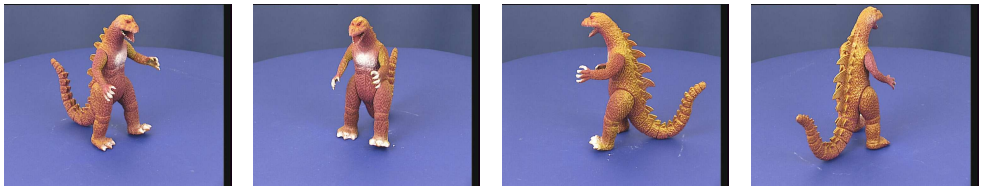


Figure 4.5: Some frames from the dinosaur sequence

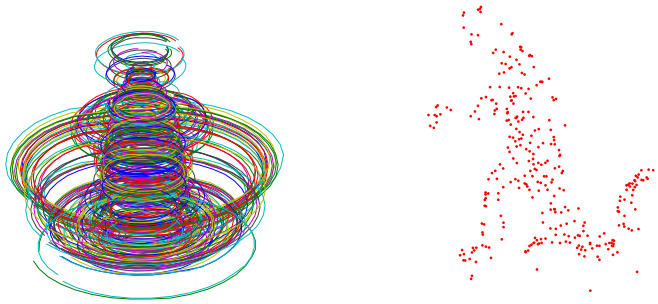


Figure 4.6: The BALM algorithm applied to the *Dinosaur* sequence. The figure on the left shows the complete 2D image trajectories as resulted from our algorithm. The figure on the right shows the 3D reconstruction of the trajectories.

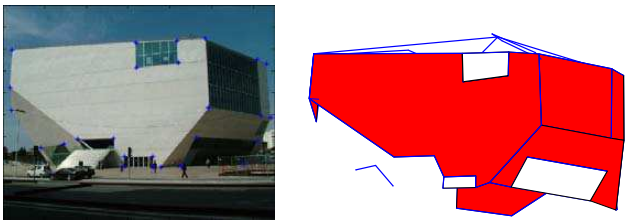


Figure 4.7: The BALM algorithm applied to the *casa da musica* images. The figure on the left shows one sample of the image set used to reconstruct the building. Note that the images have a large baseline. The figure on the right shows the 3D reconstruction of the trajectories in a pose similar to the picture on the left. Images and tracking data kindly provided by Manuel Marques.

4.7.4 Real data: Articulated SfM

We present the reconstruction of the *hinge2* sequence[5] [113]. This sequence shows two boxes linked by a hinge joint and placed on a turntable. Some frames together with the results are shown in Figure 4.8. There are 72 tracked features on the larger box and 25 features on the smaller one on top, tracked over 815 frames. We generated a random visibility matrix to simulate an amount of 60% missing data. After an initialisation obtained by filling the missing entries for the two shapes independently with rigid SfM, we apply our BALM algorithm using the projector described in the previous chapter in Section 3.2.3. The results show that even in this case of high levels of missing the data, the position of the axis is estimated correctly and it reflects the real

[5]Courtesy of Philip Tresadern.

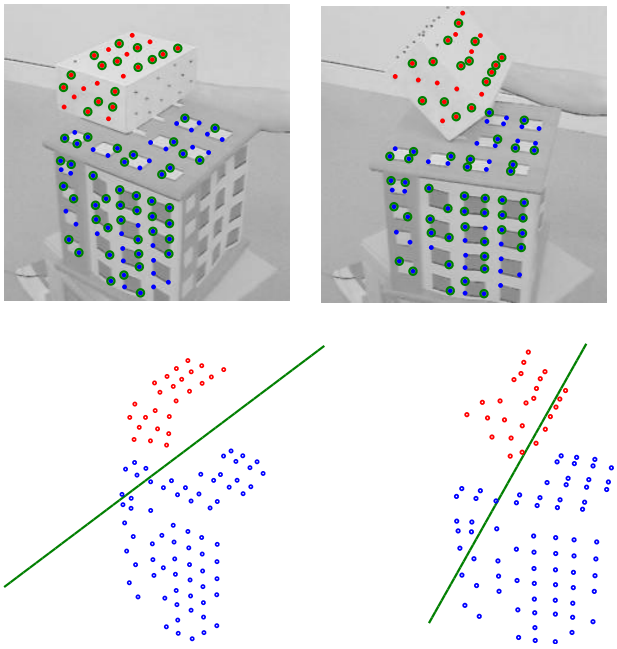


Figure 4.8: The BALM algorithm applied to the *hinge2* sequence. The figure on the top shows two samples of the image sequence and the points tracked in the image sequence. The dark green circle around some of the points represent the missing data at that given frame (60% for the whole sequence). The figures on the bottom present the 3D reconstruction together with the hinge joint localisation in 3D (green axis). Images and tracking data kindly provided by Phil Tresadern.

motion of the objects.

4.8 Summary

The BALM algorithm is a novel, general optimisation framework for a broad range of bilinear problems in Computer Vision with manifold constraints on the space where the data lies. The results demonstrated in this chapter match state of the art methods in the non-rigid and articulated structure from motion problem. The BALM method shows robustness to missing data, and the ability to solve large-scale problems. So far, we have demonstrated that optimising on the manifold of metric constraints provides robust results in spite of noise and missing data in the measurements.

All methods proposed so far in the literature, including our Metric Projections and BALM algo-

rithms, work in batch. In order for non-rigid structure from motion to replicate the popularity and success of rigid methods, a real-time method is missing. In the next chapter, we propose the first method to address the issue of sequential estimation.

Chapter 5

Sequential non-rigid structure from motion

So far the non-rigid structure from motion problem has been tackled using a batch approach. All the frames are processed at once after the video acquisition takes place. In this chapter we describe our incremental approach to the estimation of deformable models. Image frames are processed on-line in a sequential fashion. The shape is initialised to a rigid model from the first few frames. Subsequently, the problem is formulated as a model based camera tracking problem, where the pose of the camera and the mixing coefficients are updated every frame. New modes are added incrementally when the current model cannot model the current frame well enough. We define a criterion based on image reprojection error to decide whether or not the model must be updated after the arrival of a new frame. The new mode is estimated performing bundle adjustment on a window of frames. To represent the shape, we depart from the traditional explicit low-rank shape model and propose a variant that we call the 3D-implicit low-rank shape model. This alternative model results in a simpler formulation of the motion matrix and provides the ability to represent degenerate deformation modes. We illustrate our approach with experiments on motion capture sequences with ground truth 3D data and with real video sequences.

5.1 Introduction

While real-time sequential rigid SfM is a mature field that is now consolidating into commercial applications, NRSfM is still at its infancy. Some batch algorithms exist [8, 111, 85] but there is still a need to define deformable shape models and estimation algorithms that will allow to push NRSfM forward to a scenario where it might emulate the successes of its rigid counterpart, in terms of robust performance and application to real world cases. In the work we describe in this chapter we advance the state of the art in NRSfM in two main directions, both proposing a new sequential estimation paradigm and an alternative low-rank shape model.

Our first contribution is the definition of a new estimation paradigm that extends NRSfM to the sequential domain. We propose a rank-growing engine which will determine when the rank of the model should be increased and if necessary will estimate the new mode.

We divide the sequential non-rigid shape estimation into two processes: model-based tracking of the camera pose and shape coefficients and model update. The first process assumes that a current up-to-date model, of a certain rank, of the 3D shape observed so far exists and performs *model based camera tracking*: when a new frame arrives this module estimates the current camera pose and the shape parameters using as input the 2D coordinates of image features matched in the last W frames, where W is the width of a sliding window. The second process is a *model update* module which decides, based on the image reprojection error given by the camera tracking module, whether or not the current model is able to explain the deformations viewed in the new frame. If the current model does not have enough descriptive power to capture the deformations observed in the new frame, the model update module will add a new mode and estimate its parameters using bundle adjustment on a sliding window. The entire system is bootstrapped from a rigid reconstruction obtained from a small number of initial frames.

Our second contribution is an alternative low-rank shape model that provides the ability to represent modes of deformation of dimensionality lower than 3 (for instance deformations on a plane or along a line).

We call it the *3D implicit low-rank shape model* since it does not use an explicitly defined 3D

shape basis. This has two main advantages. First, the motion matrix in our model has a simpler structure than in the classical model, which allows for a linear estimation of camera pose and shape coefficients from a single frame, and can be used to initialise the bundle adjustment in the sequential framework. Second, our model handles deformations whose rank is not a multiple of 3 and thus avoids one to explicitly compute the rank of a particular shape basis. When the deformations are processed one frame at a time, having the flexibility to update the model with 1-dimensional modes fits the sequential estimation paradigm more naturally, since there is a much higher chance of observing lower dimensional deformations.

5.2 Related Work

The ability to reconstruct a deformable 3D surface from a monocular sequence when the only input information is a set of point correspondences between images is an ill posed problem unless more constraints than just the reprojection error are used. As we described in Chapter 2, current solutions to NRSfM focus on the definition of optimisation criteria to guarantee the convergence to a well behaved solution. This is often only achieved through the addition of temporal and spatial smoothness priors. Bundle adjustment has become a popular optimisation tool to refine an initial rigid solution while incorporating temporal and spatial smoothness priors on the motion and the deformations.

However, the common attribute to all NRSfM algorithms proposed so far is that they are batch methods. Our new sequential approach is motivated by recent developments in the area of sequential real-time SfM methods for rigid scenes [60, 76]. In particular, our approach is inspired by the work of Klein and Murray [60] in which they develop a real time system based on two parallel threads – the camera tracking thread which performs real time model based pose estimation and the mapping thread which runs in a constant loop performing bundle adjustment on a small set of key-frames. To the best of our knowledge our work is the first in NRSfM to depart from the batch formulation and reformulate the shape estimation sequentially. First we introduce a new variant to the low-rank linear basis shape model that we believe is better suited

to a sequential formulation.

5.3 New Deformation Model

5.3.1 Classical Explicit Low-Rank Shape Model

In the case of deformable objects the observed 3D points change as a function of time. In the low-rank shape model defined by Bregler *et al*. [15] the 3D points deform as a linear combination of a fixed set of K rigid shape bases according to time varying coefficients. In this way, $\mathtt{S}_f = \sum_{k=1}^{K} l_{fk}\mathtt{B}_k$ where the matrix $\mathtt{S}_f = [\mathbf{X}_{f1}, \cdots \mathbf{X}_{fP}]$ contains the 3D coordinates of the P points at frame f, the $3 \times P$ matrices $\mathtt{B}_k$ are the shape bases and l_{fk} are the coefficient weights. If the 3D shape is known, this model can be obtained from the PCA decomposition of the $\mathtt{S}^*$ that contains the 3D shape in all the frames.

$$\mathtt{S}^*_{F\times 3P} = \begin{bmatrix} \mathtt{S}^*_1 \\ \mathtt{S}^*_2 \\ \vdots \\ \mathtt{S}^*_F \end{bmatrix} = \begin{bmatrix} X_{11} & Y_{11} & Z_{11} & \cdots & X_{1P} & Y_{1P} & Z_{1P} \\ & \vdots & & & & \vdots & \\ X_{F1} & Y_{F1} & Z_{F1} & \cdots & X_{FP} & Y_{FP} & Z_{FP} \end{bmatrix} \tag{5.1}$$

A PCA decomposition of rank K of $\mathtt{S}^*$ would give $\mathtt{LB}^*$, where $\mathtt{L}$ is the $F \times K$ matrix of deformation weights l_{ik}, and the $K \times 3P$ matrix $\mathtt{B}^*$ can be rearranged to give the basis shapes $\mathtt{B}_k$. If we assume an orthographic projection model the coordinates of the 2D image points observed at each frame i are then given by:

$$\mathtt{W}_i = \mathtt{R}_i \left(\sum_{k=1}^{K} l_{ik}\mathtt{B}_k \right) + \mathtt{T}_i \tag{5.2}$$

where $\mathtt{R}_i$ is a 2×3 *Stiefel matrix* and $\mathtt{T}_i$ aligns the image coordinates to the image centroid. The aligning matrix $\mathtt{T}_i$ is such that $\mathtt{T}_i = \mathbf{t}_i \mathbf{1}_P^T$ where the 2-vector $\mathbf{t}_i$ is the 2D image centroid and $\mathbf{1}_P$ a vector of ones.

When the image coordinates are registered to the centroid of the object and we consider all the

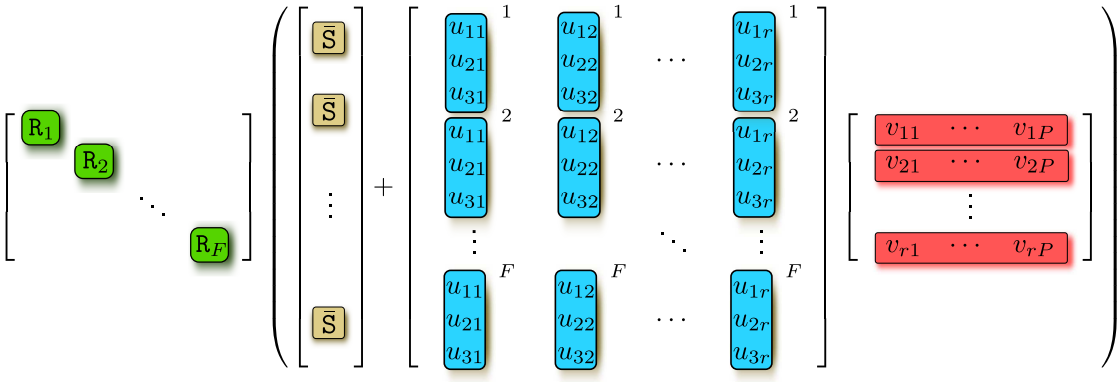


Figure 5.1: The proposed 3D-Implicit Low-Rank Shape Model, the camera matrices for all frames R_f are collected in a block-diagonal projection matrix, the time-varying 3D shape is represented by a $3F \times P$ structure, modelled by the sum of a rigid component $\bar{\mathtt{S}}$ and a rank-r decomposition of the non-rigid component.

frames in the sequence, we may write the measurement matrix as:

$$\mathtt{W} = \begin{bmatrix} l_{11}\mathtt{R}_1 & \dots & l_{1K}\mathtt{R}_1 \\ \vdots & \ddots & \vdots \\ l_{F1}\mathtt{R}_F & \dots & l_{FK}\mathtt{R}_F \end{bmatrix} \begin{bmatrix} \mathtt{B}_1 \\ \vdots \\ \mathtt{B}_K \end{bmatrix} = \mathtt{MS} \tag{5.3}$$

Since $\mathtt{M}$ is a $2F \times 3K$ matrix and $\mathtt{S}$ is a $3K \times P$ matrix in the case of deformable structure the rank of $\mathtt{W}$ is constrained to be at most $3K$. The motion matrices now have a complicated repetitive structure $\mathtt{M}_i = [\mathtt{M}_{i1} \dots \mathtt{M}_{iK}] = [l_{i1}\mathtt{R}_i \dots l_{iK}\mathtt{R}_i]$ that makes the model estimation difficult.

Olsen and Bartoli [83] proposed to consider an implicit model where the repetitive structure of the motion matrix is not used. While this simplifies the estimation problem, the recovered model does not directly provide usable motion and shape parameters, unless a mixing matrix is computed [15, 128]. The mixing matrix computation problem has not received a simple solution so far.

5.3.2 Proposed 3D-Implicit Low-Rank Shape Model

We propose a way to depart from the traditional basis shapes model, and embrace a different formulation that will fit the problem of sequential structure recovery more naturally since it allows for the rank of the shape model to grow one by one with the arrival of a new frame, instead of multiples of three.

The data in the shape matrix may be re-arranged in a different form, stacking the shape matrices vertically for all frames F. Each matrix $\mathtt{S}_f \in \mathbb{R}^{3\times P}$ contains the 3D coordinates of P points in frame f.

$$\mathtt{S}_{3F\times P} = \begin{bmatrix} \mathtt{S}_1 \\ \mathtt{S}_2 \\ \vdots \\ \mathtt{S}_F \end{bmatrix} = \begin{bmatrix} X_{11} & X_{12} & & X_{1P} \\ Y_{11} & Y_{12} & \cdots & Y_{1P} \\ Z_{11} & Z_{12} & & Z_{1P} \\ \vdots & \vdots & & \vdots \\ X_{F1} & X_{F2} & & X_{FP} \\ Y_{F1} & Y_{F2} & \cdots & Y_{FP} \\ Z_{F1} & Z_{F2} & & Z_{FP} \end{bmatrix} \tag{5.4}$$

If we assume that the shape matrix $\mathtt{S}$ is low-rank we can perform Principal Components Analysis to obtain a PCA basis as $\mathtt{S} = \mathtt{U}_d\mathtt{V}_d$, where d is the rank of the decomposition, $\mathtt{U}_d \in \mathbb{R}^{3F\times d}$ and $\mathtt{V}_d \in \mathbb{R}^{d\times P}$. We can also explicitly include an average rigid (mean) shape in the model, therefore the shape at frame f would be given by:

$$\mathtt{S}_f = \bar{\mathtt{S}} + \begin{bmatrix} \mathbf{U}_{f1} & \cdots & \mathbf{U}_{fr} \end{bmatrix} \begin{bmatrix} \mathbf{V}_1 \\ \mathbf{V}_2 \\ \vdots \\ \mathbf{V}_r \end{bmatrix} \tag{5.5}$$

where $\bar{\mathtt{S}}$ is the mean shape, $d = 3+r$, $\mathbf{U}_{fr}$ is the 3-vector $[U(x)_{fr}U(y)_{fr}U(z)_{fr}]^T$ and $\mathbf{V}_r$ are the rows of matrix $\mathtt{V}$.

Therefore we can consider $\mathtt{V}$ to be a PCA basis of the shape (row) space of $\mathtt{S}$ and $\mathtt{U}$ to contain the time varying coefficients. Note that in this case the shape matrix $\mathtt{V}$ has dimensions $r \times P$ where r is the rank of the decomposition and P is the number of points in the shape. For each frame $3r$ coefficients are needed to express the configuration of the shape.

We assume that the shape at instant f is then projected onto an image following an orthographic camera model. The 2D coordinates of the points can then be expressed as:

$$\mathtt{W}_f = \begin{bmatrix} u_{f1} & \cdots & u_{fP} \\ v_{f1} & \cdots & v_{fP} \end{bmatrix} = \mathtt{R}_f \mathtt{S}_f + \mathbf{T}_f = \mathtt{R}_f(\bar{\mathtt{S}} + \mathtt{U}_f \mathtt{V}) + \mathbf{T}_f \tag{5.6}$$

where $\mathtt{R}_f$ is a $[2 \times 3]$ orthographic camera projection matrix, it encodes the first two rows of the camera rotation matrix and $\mathtt{T}_f$ the translation for frame f. If we now register all the measurements to their centroid in each frame the projection of the shape in all frames can be written as:

$$\mathtt{W} = \begin{bmatrix} \mathtt{R}_1 & & & \\ & \mathtt{R}_2 & & \\ & & \ddots & \\ & & & \mathtt{R}_F \end{bmatrix} \left(\begin{bmatrix} \bar{\mathtt{S}} \\ \bar{\mathtt{S}} \\ \vdots \\ \bar{\mathtt{S}} \end{bmatrix} + \begin{bmatrix} \mathbf{U}_{11} & \cdots & \mathbf{U}_{1r} \\ \mathbf{U}_{21} & \cdots & \mathbf{U}_{2r} \\ \vdots & & \vdots \\ \mathbf{U}_{F1} & \cdots & \mathbf{U}_{Fr} \end{bmatrix} \begin{bmatrix} \mathbf{V}_1 \\ \mathbf{V}_2 \\ \vdots \\ \mathbf{V}_r \end{bmatrix} \right) \tag{5.7}$$

A visual representation of this new model can be seen in Figure 5.2. The results from the experiment in section 5.9.2 are used to display an average 3D deformation across all frames in the sequence. Each image in Figure 5.2 corresponds to the effect of one element of the $\mathtt{U}$ matrix. Each row affects one of the coordinates, and each column is related to the rank-r basis contained in $\mathtt{V}$. Note that the basis are not independent, and some might be zero. Also, the incremental nature of our method (as explained in the following sections), is such that a rank-1 base in the matrix $\mathtt{V}$ can encode stronger deformations than the previous basis. In our model, the basis shapes are not explicitly used as in the classical model, while the camera projection is explicitly modelled. We thus call our model the *3D-implicit low-rank shape model*. Our model combines Bregler *et al.* [15]'s explicit model and Olsen and Bartoli [83]'s implicit model. It has

the following two main advantages:

1. **Simplicity.** The motion matrix is block diagonal and only contains the rotation matrices instead of a mixture of the coefficients and the rotations. The fact that the 3D basis is not explicitly available in our model is not a problem since one is generally more interested in recovering the 3D shape of the observed scene than the basis shapes – the basis shapes can be estimated a posteriori by forming and factorising the matrix $\mathtt{S}^*$ in equation (5.1). As we explain below, it also is an advantage not to have explicit 3D basis shapes.

2. **Any-rank deformations.** Our formulation allows us to define shape models where the rank is not a multiple of 3. In other words, in the explicit model, a basis shape always has to be of rank 3, whereas in the real world not all deformations are of rank 3. Xiao and Kanade [129] propose to explicitly find the rank of a particular deformation mode (which can be one of 1, 2 or 3). Our model circumvents this difficult problem.

5.4 A Sequential Approach to NRSfM

In this work we depart from the batch formulation of NRSfM and we propose a sequential approach based on the alternative low-rank shape model outlined in the previous section. Our approach can be seen as a two process formulation. The system holds a current up-to-date model, of a certain rank, encapsulated in matrix $\mathtt{V}$. The first process is a model based camera tracking module. Given the current estimate of $\mathtt{V}$, when a new frame arrives, the camera tracking module estimates the new pose $\mathtt{R}_f$ and the new deformation coefficients $\mathtt{U}_f$ for the current frame. If the current model explains well the measurements the image reprojection error will be low. However, if the error goes above some defined threshold the rank of the model must be increased and the model updated. In that case, a model update module will update the current model adding a new row to matrix $\mathtt{V}$. As the sequence is processed the model will become more complicated, until all the possible object deformations have been observed. Our sequential

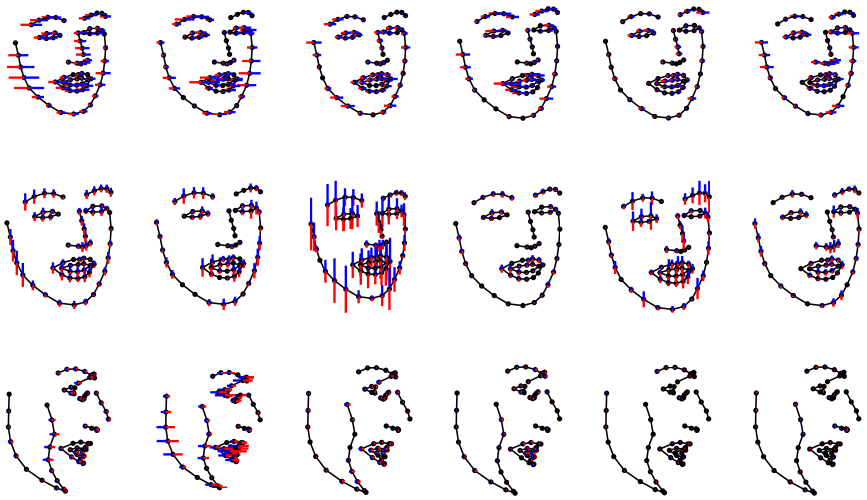


Figure 5.2: A visual representation of the 3D implicit model coefficients: Each image shows the contributions to points deformations given by each element of the $3 \times r$ matrix $\mathtt{U}_f$. Each row of the $\mathtt{U}_f$ matrix affects one coordinate, and each column is the weight to be given to the corresponding row of the $\mathtt{V}$ matrix, each row representing an added rank-1 mode of deformation.

approach to NRSfM is summarised in Algorithm 5. We now describe in detail the two main modules of our sequential system: the camera tracking module and the model update module.

5.5 Camera Tracking Given a Known Model V

If the matrix $\mathtt{V}$ is known in advance, the NRSfM problem is reduced to the estimation of the camera pose $\mathtt{R}_f$ and the mixing coefficients $\mathtt{U}_f$ for each frame. In that case, the pose of the camera and the coefficients can be updated sequentially for each frame using a model based approach.

We adopt a sliding window approach where we perform bundle adjustment on the last N frames where N is the width of a pre-defined window. The cost to be minimised is the image reprojection error over all frames in the window:

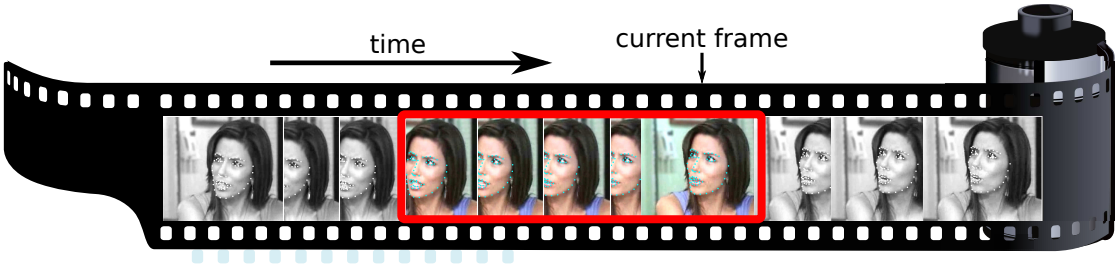


Figure 5.3: Sliding window approach: A group of frames (shown in colour) is processed at each step. This keeps the computational cost bounded. As a new frame becomes available, the group of frame "slides", and the new frame is processed. At each step the 3D shape and camera motion of the new frame is computed.

Algorithm 5 Sequential non-rigid structure from motion (NRSfM)

Require: 2D point correspondences

Ensure: 3D coordinates of the deforming surface for each frame.

1: Initialise model to mean rigid shape $\bar{\mathtt{S}}$ estimated via rigid factorisation on the first few frames.
2: **loop**
3: new frame f arrives
4: run *camera tracking process*: estimate camera pose $\mathtt{R}_i$ and coefficients $\mathtt{U}_i$
5: **while** (image reprojection error is above threshold) **do**
6: run *model update process*:
7: increase rank $r \leftarrow r+1$
8: estimate new row of $\mathtt{V}$ and new column of $\mathtt{U}_f$
9: **end while**
10: go to process next frame; $f \leftarrow f+1$
11: **end loop**

$$\min_{\mathtt{R}_i,\mathtt{U}_i} \sum_{i=f-N}^{f} \|\mathtt{W}_i - \mathtt{R}_i(\bar{\mathtt{S}} + \mathtt{U}_i\mathtt{V})\|_F^2 \tag{5.8}$$

To this cost function we add a temporal smoothness prior to penalise strong variations in the camera matrices of the form $\|\mathtt{R}_i - \mathtt{R}_{i-1}\|_F^2$, and a shape smoothness prior (similar to the one used in [8]) that ensures that points that lie close to each other in space should stay close. The shape smoothness is defined as $\sum_{i=f-N}^{f} D^{i,i-1}$, where $D^{i,i-1}$ is the change in the euclidean distance between 3D points over two frames: $D^{i,i-1} = \sum_{a,b=1}^{P} \phi_{a,b} |d^2(\mathbf{X}_{i,a}, \mathbf{X}_{i,b}) - d^2(\mathbf{X}_{i-1,a}, \mathbf{X}_{i-1,b})|$. The weight $\phi_{a,b}$ is a measure of the closeness of points a and b, defined as a $P \times P$ affinity matrix

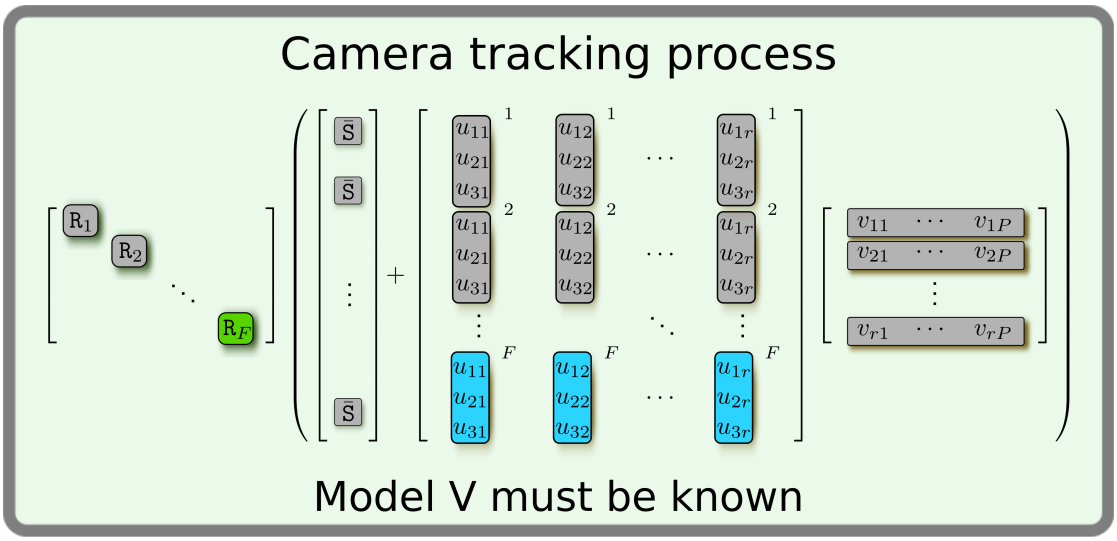


Figure 5.4: Camera tracking process: Assuming a known model for the deformations and a known mean shape, the camera matrix and deformation coefficients for only one frame are estimated.

$\phi_{a,b} = \rho(d^2(\mathbf{X}_a, \mathbf{X}_b))$ where ρ is a truncated Gaussian kernel. The final cost function can now be written as:

$$\min_{\mathtt{R}_i,\mathtt{U}_i} \sum_{i=f-N}^{f} \|\mathtt{W}_i - \mathtt{R}_i(\bar{\mathtt{S}} + \mathtt{U}_i\mathtt{V})\|_F^2 + \lambda \sum_{i=f-N}^{f} \|\mathtt{R}_i - \mathtt{R}_{i-1}\|_F^2 + \psi \sum_{i=f-N}^{f} D^{i,i-1} \tag{5.9}$$

The mean shape $\bar{\mathtt{S}}$ and the shape model $\mathtt{V}$ are assumed to be known. This nonlinear minimisation requires an initial estimate for the camera pose $\mathtt{R}_f$ and the shape coefficients $\mathtt{U}_f$ in the current frame f. Algorithms to obtain linear estimates for $\mathtt{R}_f$ and $\mathtt{U}_f$ are described in Section 5.5.1.

The steps of the complete algorithm to track the current pose of the camera and the shape coefficients given the shape model can be summarised as follows. Each time a new frame f of feature tracks is available:

- Obtain initial estimates for the current pose $\mathtt{R}_f$ and mixing coefficients $\mathtt{U}_f$ using the linear estimation plus prior described in Section 5.5.1.

- Minimise the cost function (5.9) with smoothness priors using bundle adjustment to obtain

optimised values for the rotations $\mathtt{R}_i$ and shape coefficients $\mathtt{U}_i$ in all the frames in the sliding window.

- If the reprojection error of the window becomes higher than a threshold, signal the modelling process to increase the rank of the $\mathtt{V}$ matrix.

5.5.1 Initialisation: Linear Estimation of $\mathtt{U}_f$ and $\mathtt{R}_f$

Consider new image measurements become available for a new frame. These can be arranged in a $2 \times P$ matrix for that single frame called $\mathtt{W}_f$. The projection model gives us the relation $\mathtt{W}_f = \mathtt{R}_f(\bar{\mathtt{S}} + \mathtt{U}_f\mathtt{V}) + \mathtt{T}_f$.

Linear estimation of $\mathtt{R}_f$.

For every new frame the camera pose $\mathtt{R}_f$ must be initialised before Bundle Adjustment. For this purpose, we approximate the shape with the rigid mode to obtain an initial estimate of the camera rotation. This means we need to find the camera pose $\mathtt{R}_f$ that satisfies $\mathtt{W}_f = \mathtt{R}_f\mathtt{S}$, while respecting the smoothness prior $\lambda\,\mathtt{I}\,\mathrm{vec}(\mathtt{R}_f) = \lambda\,\mathrm{vec}(\mathtt{R}_{f-1})$. Using the relation $\mathrm{vec}(\mathtt{AXB}) = [\mathtt{B}^T \otimes \mathtt{A}]\,\mathrm{vec}(\mathtt{X})$, where $\otimes$ is the Kronecker product and $\mathrm{vec}(.)$ is the column-major vectorisation of a matrix, and using $\mathtt{W}_f = \mathtt{I}_2\mathtt{R}_f\mathtt{S}$ we can write:

$$\mathrm{vec}(\mathtt{W}_f) = [\mathtt{S}^T \otimes \mathtt{I}_2]\,\mathrm{vec}(\mathtt{R}_f) \tag{5.10}$$

$$\begin{bmatrix} [\mathtt{S}^T \otimes \mathtt{I}_2] \\ \lambda\,\mathtt{I} \end{bmatrix} \mathrm{vec}(\mathtt{R}_f) = \begin{bmatrix} \mathrm{vec}(\mathtt{W}_f) \\ \lambda\,\mathrm{vec}(\mathtt{R}_{f-1}) \end{bmatrix} \tag{5.11}$$

The resulting $\mathtt{R}_f$ will not be orthonormal (i.e. not a truncated rotation matrix), so we find the closest orthonormal rigid projection using SVD.

Linear estimation of $\mathtt{U}_f$.

First we take away the contribution to the image measurements given by the known translation and mean shape component to give $\tilde{\mathtt{W}}_f = \mathtt{W}_f - \mathtt{T}_f - \mathtt{R}_f\bar{\mathtt{S}} = \mathtt{R}_f\mathtt{U}_f\mathtt{V}$, which can be rewritten as

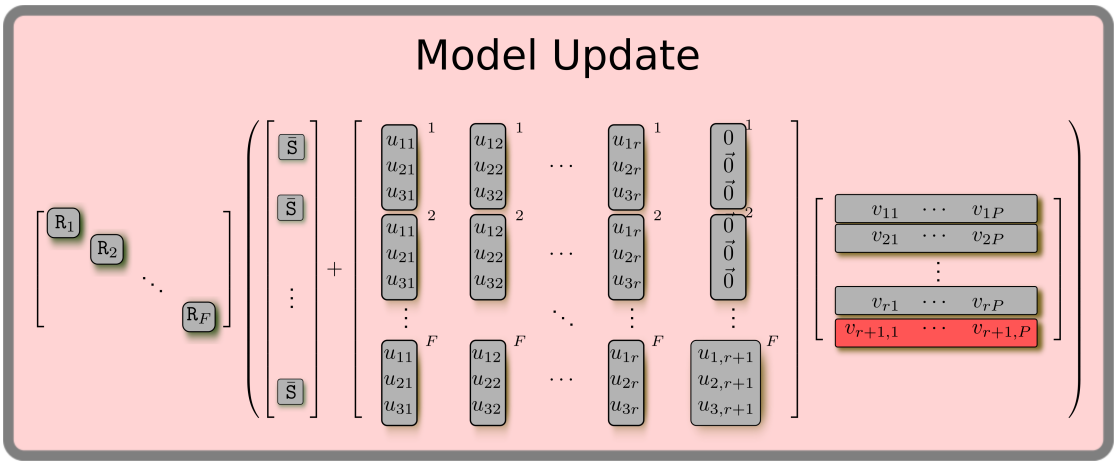


Figure 5.5: Model update process: a new deformation basis is added to the V matrix, increasing the rank of the deformation model. More deformations can be expressed with a model of increased complexity, hence the deformation coefficients can be re-estimated to match the input data.

$\text{vec}(\tilde{\mathtt{W}}_f) = [\mathtt{V}^T \otimes \mathtt{R}_f]\,\text{vec}(\mathtt{U}_f)$. This provides a linear equation on the unknown vector $\mathtt{U}_f$. However, this is not sufficient to produce an acceptable solution, because $\mathtt{U}_f$ is a $3 \times r$ matrix where each column $\mathbf{U}_{fr}$ is a 3-vector $[U(x)_{fr}U(y)_{fr}U(z)_{fr}]^T$ that contains the PCA coefficients of all 3D coordinates, while $\tilde{\mathtt{W}}_f$ contains 2D projections. However, this problem can be overcome by including a temporal smoothness prior term that penalises solutions that are far from the value for the previous frame $\mathtt{U}_{f-1}$. Thus the prior term is of the form $\lambda\,\mathtt{I}\,\text{vec}(\mathtt{U}_f) = \lambda\,\text{vec}(\mathtt{U}_{f-1})$. We can join both linear equations and solve the linear system:

$$\begin{bmatrix} [\mathtt{V}^T \otimes \mathtt{R}_f] \\ \lambda\,\mathtt{I} \end{bmatrix} \text{vec}(\mathtt{U}_f) = \begin{bmatrix} \text{vec}(\tilde{\mathtt{W}}_f) \\ \lambda\,\text{vec}(\mathtt{U}_{f-1}) \end{bmatrix} \tag{5.12}$$

5.6 Sequential Update of the Shape Model

In NRSfM the shape of the 3D object the camera observes varies over time. The current model will encode the modes of deformation that the object has exhibited so far in the sequence. However, if the object deforms in different ways that are not encoded in the model the camera tracking will fail. Therefore, a mechanism is needed to update the model when new modes of deforma-

tion appear. In that case, the rank of the model should grow and the parameters of the model should be fit to the new data.

The difficulty of updating the model in an sequential way is double-fold. Firstly, when each new frame arrives, we need a mechanism to decide whether or not the current model continues to fit the data well enough. While the shape model can still describe the data, we can continue to do model based camera tracking. We decide this based on the image reprojection error. Secondly, if the model can no longer explain the data, the rank of the model needs to grow to incorporate the new mode of deformation and the parameters of the new row of `V` and the new column of `U` must be estimated.

5.6.1 Rank Increase Criterion

The rank selection criterion will decide to increase the rank only if the current data does not fit the model well enough, i.e. if the existing modes do not model the current frame well. Therefore we use the image reprojection error as the criterion – if the error increases above a certain threshold we increase the rank of the shape model. This results in a new row being added to the PCA basis `V` and a new column to the PCA components `U`. However, the new mode is recovered from the current frame only, so it has no influence over past frames. Therefore for all past frames we can set the $3(f-1)$ components of the new column of `U` to 0.

5.6.2 Model Update: Estimating New Row of `V` and New Column of `U`

When the camera tracking module processes a new frame that it cannot model well enough (the reprojection error is above the defined threshold), the model is updated by increasing the rank. Ideally once all the different modes of deformation that an object can exercise are incorporated in the PCA basis, the rank will remain stable and the camera tracking process will be able to reconstruct the incoming frames.

Given new image correspondences for frame f, the rank of `U`,`V` must be increased. From the

current estimate of $\mathtt{U}_{f,1:r-1}$ and $\mathtt{V}_{1:r-1}$ we can rewrite the model for the new frame as

$$\tilde{\mathtt{W}}_f = \mathtt{R}_f(\bar{\mathtt{S}} + \mathtt{U}_{f,1:r-1}\mathtt{V}_{1:r-1} + \mathtt{U}_{f,r}\mathtt{V}_r). \tag{5.13}$$

Both the residual of the current model $\mathtt{A} = \tilde{\mathtt{W}}_f - \mathtt{R}_f\,(\bar{\mathtt{S}} + \mathtt{U}_{f,1:r-1}\mathtt{V}_{1:r-1})$ and the current camera rotation $\mathtt{R}_f$ are known. We need to estimate $\mathtt{Z} = \mathtt{U}_{f,r}\mathtt{V}_r$, the contribution of the new rank, subject to the following constraints:

$$\mathtt{A} = \mathtt{R}_f\mathtt{Z} \quad \text{rank}(\mathtt{Z}) = 1 \tag{5.14}$$

This problem is difficult to solve in closed form, therefore we approximate it using a linear solution as follows. We define $\mathtt{C}$ as the closest rank-1 approximation of $\mathtt{A}$ obtained using SVD, then compute $\mathtt{Z}$ as $\mathtt{Z} = \mathtt{R}_f^{\dagger}\mathtt{C}$. Finally, we can decompose $\mathtt{Z}$ using a rank-1 SVD decomposition to obtain a new row for $\mathtt{V}$.

Non-linear refinement

Once initial estimates are available for the new row of $\mathtt{V}$ and the new column of $\mathtt{U}$, they can be refined minimising image reprojection error over a sliding window of N frames

$$\min_{\mathtt{V}_r,\mathtt{U}_{ir}} \sum_{i=f-N}^{f} \|\mathtt{W}_i - \mathtt{R}_i(\bar{\mathtt{S}} + \mathtt{U}_i\mathtt{V})\|_F^2 \tag{5.15}$$

incorporating the smoothness priors described in section 5.5. Once the model is updated, the camera tracking module can resume *model based tracking* with the new model $\mathtt{V}$ with rank $r+1$.

5.6.3 Bootstrapping

One of the known challenges in sequential approaches to rigid SfM is the initialisation [60]. It is common to run the system in batch mode for a few frames to obtain a first model of the scene before starting the sequential operation. In the current experiments we run a rigid factorisation algorithm on a few initial frames to obtain the rigid mean shape $\bar{\mathtt{S}}$. Once this is available the camera tracking and model update loop can start. An alternative approach that does not require

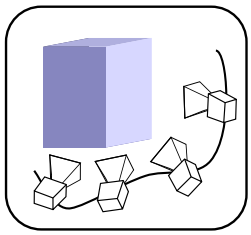


Figure 5.6: At the beginning, the camera tracking process requires an average rigid shape. This can be acquired using a subset of the sequence where the object is rigid. For example, if the beginning of the video shows rigid motion, the initial frames can be used.

manual intervention is the following. Start performing rigid factorisation in batch. When a new frame arrives, if the reprojection error of rigid factorisation over the frames observed so far is below the threshold then we keep performing rigid factorisation. However, if the error becomes higher than our threshold, the mean shape of the non-rigid model is set to the rigid model obtained so far and we start our sequential NRSfM algorithm.

5.7 Limiting the rank

Section 5.6.2 described the rank-growing engine that allows us to learn a 3D deformable model in a frame-by-frame fashion. The current formulation does not impose an upper bound on the total rank of the model which could grow without limit. Sequential SfM methods, however, rely on the computational complexity remaining bounded. The complexity of the camera tracking process depends quadratically on the number of unknown parameters which grows linearly with the rank of the decomposition. Therefore, we must incorporate a mechanism to compact the model in order to limit its overall rank. This is particularly useful when dealing with longer sequences. For this purpose, we add a user-specified limit to the rank of the non-rigid decomposition, and use PCA to compress the rank of the model when it grows above the threshold.

5.7.1 Model compression

Subtracting the average rigid shape $\bar{\mathtt{S}}$ from the shape model $\mathtt{S}$, we can express the remaining non-rigid component as the product of two low-rank matrices $\mathtt{U}$ and $\mathtt{V}$ of rank r:

$$\mathtt{S} - \bar{\mathtt{S}} = \mathtt{UV} \tag{5.16}$$

When the rank of the non-rigid component reaches the limit r, we apply PCA to truncate the decomposition to rank $\frac{r}{2}$. In this way, we keep the computational complexity bounded since the rank of the model cannot grow beyond the user-specified limit.

5.8 Missing data

The need to adapt this technique to the case of missing data is clear — for each frame we must be able to deal with occlusions and lost tracks. Bundle adjustment has the built-in capability to deal with missing data since only the visible points in each frame are evaluated in the cost function:

$$\min_{\mathtt{R}_i, \mathtt{U}_i} \sum_{i=f-N}^{f} \sum_{j \in \mathcal{O}} |\mathbf{W}_{ij} - \mathtt{R}_i(\bar{\mathtt{S}}_j + \mathtt{U}_i \mathbf{V}_j)|^2 \tag{5.17}$$

where $\mathcal{O}$ is the set of observable data points. In this way, provided the amount of known data is larger than the number of parameters to estimate, the camera tracking problem can be solved in the presence of missing data.

Regarding the model-update module, the formulation described in Section5.6.2, assumed full data. When the tracking data in the current frame contains occlusions, we restrict the calculation only to the known points:

$$\tilde{\mathtt{W}}_{ij} = \mathtt{R}_i(\bar{\mathtt{S}_j} + \mathtt{U}_{f,1:r-1}\mathtt{V}_{(1:r-1,j)} + \mathtt{U}_{f,r}\mathtt{V}_{rj}). \tag{5.18}$$

Stacking equation 5.18 horizontally for all the observable points $j \in \mathcal{O}$, we can obtain an update for the $\mathtt{V}$ matrix. We fill the entries of the new row of the shape model matrix $\mathtt{V}_r$ associated with

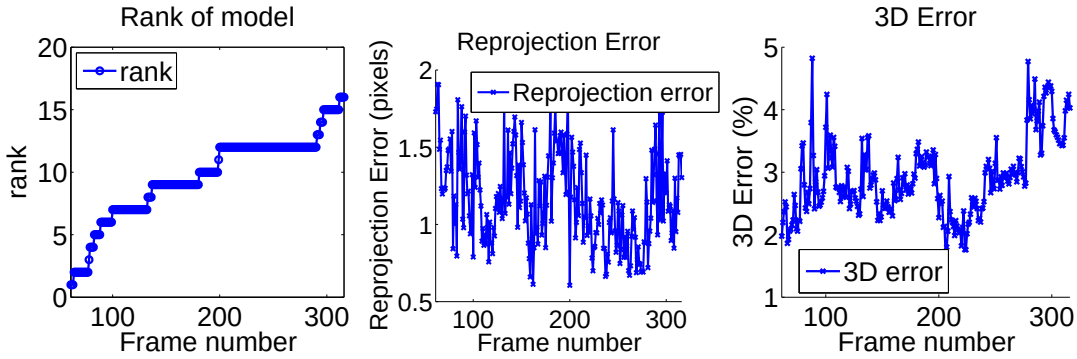


Figure 5.7: Results of sequential NRSfM on the CMU-face sequence. Left: Value of the rank of the model for each frame, increasing as more frames are processed. Middle: 2D Reprojection error given by the camera tracking process. Right: 3D error of the reconstruction for each frame.

the missing data points with zeroes, implicitly assuming that those points do not contribute to the new mode of deformation.

This solution to camera tracking and model update with missing data is demonstrated experimentally in section 5.9.3.

5.9 Experiments

5.9.1 Motion capture sequence *CMU-face*

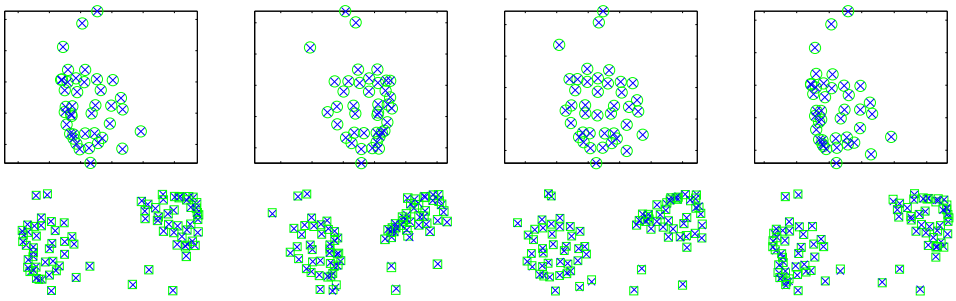
First we tested our sequential method based on the 3D-implicit low-rank shape model on a motion capture sequence with ground truth data[1]. This sequence from the CMU Motion Capture Database[2] contains 316 frames of motion capture data of the face of a subject wearing 40 markers performing deformations while rotating. This sequence was also used by Torresani *et al*. [111] to perform quantitative tests with ground truth data. We projected the 3D data synthetically using an orthographic camera model.

Prior to the start of our sequential algorithm and with the purpose of bootstrapping the camera tracking module, we ran a batch rigid SfM algorithm [110] on the first 60 frames of the sequence

[1] Videos of the experimental results can be found on the project website http://www.eecs.qmul.ac.uk/~lourdes/SequentialNRSFM

[2] Available from http://mocap.cs.cmu.edu

to estimate the mean shape $\bar{\mathbf{S}}$. The PCA basis matrix $\mathtt{V}$ was initialised to $\mathtt{0}$. We then ran our new sequential algorithm based on the camera tracking and the model update modules, together with the rank detection engine. The average 3D error is 2.9%, with a 0.7 pixels 2D reprojection error on the 600×600pixels images. The reprojection threshold was fixed to 1.2pixels.



Frame 61 Frame 188 Frame 252 Frame 316

Figure 5.8: 3D Reconstruction results obtained on the *CMU-face* sequence using camera tracking and model updating. First row: 2D image points (green circles) and reprojections (blue crosses). Second row: Views of the 3D reconstruction (crosses) compared with ground truth MOCAP data (squares)

In Figure 5.7 we show results of the rank estimation, the 2D image reprojection error and the 3D error for each frame in the sequence using our sequential estimation formulation. The average image reprojection error over the whole sequence is less than a pixel. In Figure 5.9 (left) we compare results of the 3D error obtained with our method (Sequential), with Torresani *et al.*'s state of the art batch NRSfM algorithm (EM-LDS) [111].We show the histogram of 3D error values taking into account all the frames in the sequence. The results show that our new sequential algorithm provides results comparable to Torresani *et al.*'s [111] batch state of the art algorithm. We show smooth estimates of the rotation angles for all the frames in the sequence in Figure 5.9 (right). In Figure 5.8 we show the 2D image reprojection error and the 3D reconstructions (blue crosses) we obtained for some frames in the sequence comparing them with ground truth values (green squares).

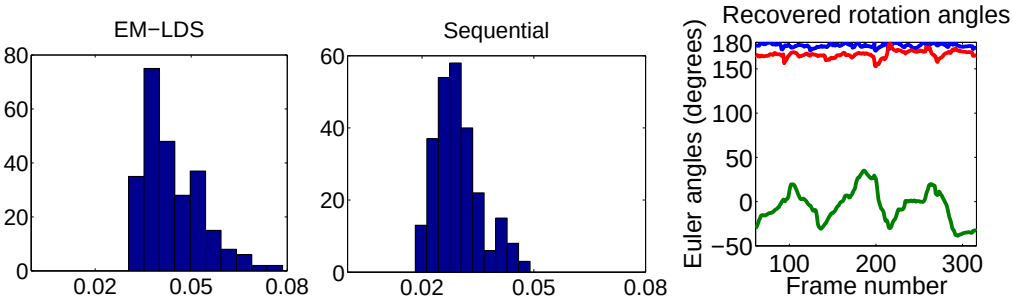


Figure 5.9: (Left) Histogram of 3D error values built from all the frames, comparing results of our method (Sequential) with Torresani *et al.*'s state of the art batch (EM-LDS) [111]. The 3D errors obtained with our Sequential approach are comparable to the results from the batch method EM-LDS. (Right) Rotation angles estimated with the camera tracking module.

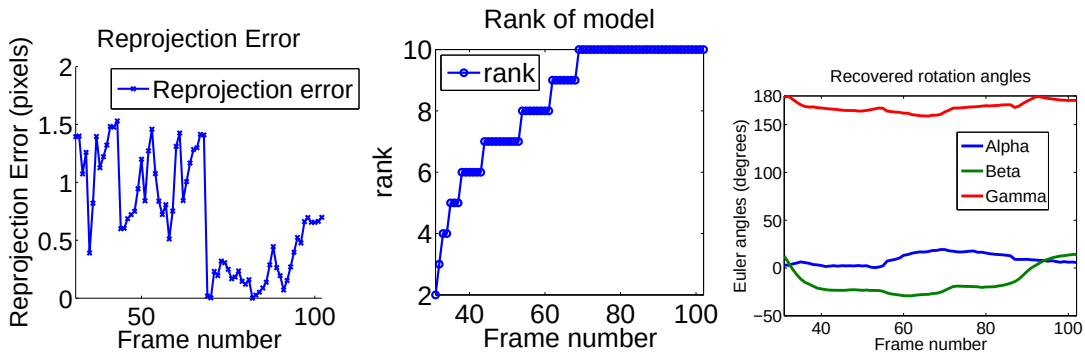


Figure 5.10: Results on the *actress* sequence. Left: Reprojection error of the frame-by-frame reconstruction obtained with our method. Middle: The value of the rank, increased as more frames are processed. Right: Rotation angles estimated with the camera tracking module.

5.9.2 Real Data

We used the *actress* sequence, also used by Bartoli *et al.* [8], which consists of 102 frames of a video showing an actress talking and moving her head. In Figure 5.11 we show results of the 3D reconstructions obtained for some of the frames in the sequence. The camera tracking was bootstrapped with a rigid model obtained using Tomasi and Kanade's rigid factorisation algorithm [110] on the first 30 frames. The threshold for increasing the rank was a reprojection error of 0.9 pixels. From figure 5.10 we can see that the rank is increased, and the estimation of new frame parameters keeps the reprojection error low.

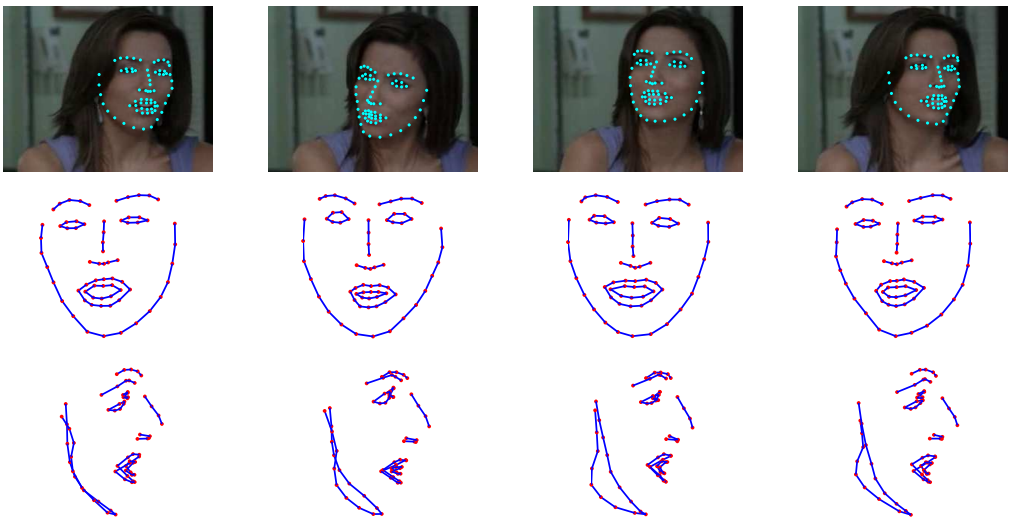


Figure 5.11: Qualitative results on the *actress* sequence using camera tracking and model update. First row: The input images with superimposed feature tracking data. Second and Third rows: Front and side views of the 3D reconstruction of 4 frames of the sequence.

5.9.3 Missing data

We used a real sequence with occlusions (kindly provided by P. Gotardo and A. Martinez [45]) of a person performing American sign language gestures. The sequence is 114 frames long, and the 77 markers on the face were manually tracked in all frames where they were visible. The features are often occluded in this sequence due to hand gestures and self occlusions. Figure 5.13 shows the results we obtain in a sequential estimation, highlighting the recovery of missing data. Deformations are correctly recovered, and the overall rms reprojection error is 1 pixel. For this sequence we used the model compression method described in section 5.7, imposing a limit on the rank of the decomposition to rank 12. Figure 5.12 shows the reprojection error for each frame, the rank of the recovered model, and the missing data visibility matrix.

5.10 Application to Model-based feature tracking

Once acquired, a 3D deformable shape model is a general representation of an object which can subsequently be used for tracking. Our sequential modelling algorithm [84] can indeed

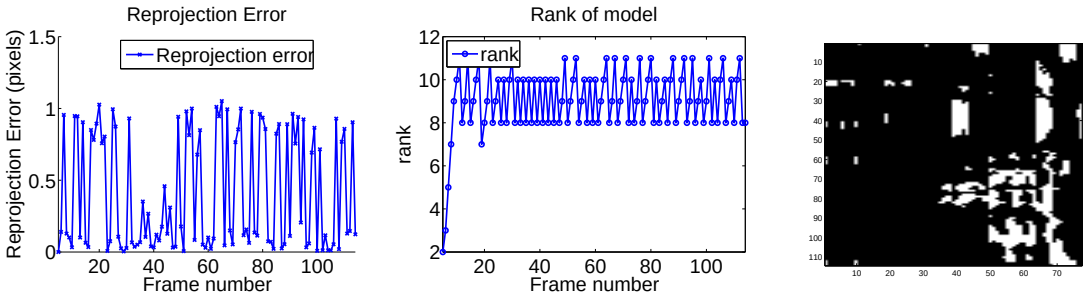


Figure 5.12: Sequential estimation of the sign language sequence. Left: reprojection error, Middle: Rank of the decomposition, showing model compression. Right: Missing data in the sequence, black points are observable features, missing data in white.

produce from scratch such models. Model-based tracking can then be defined as the task of identifying the pose and deformation coefficients of the object for each frame of a stream of further images. If correctly formulated, tracking with a known 3D model can be performed sequentially, proceeding from step to step based only on a current state estimate and new image data and without needing to refer back to older images. This is usually achieved by combining the current image measurements with the reprojection of the predicted model combined with priors on the parameters of the model.

Tracking non-rigid objects using a 3D model is an active research area, particularly in the case of human faces due to its applications to computer graphics animation, human computer interaction or face recognition. Most approaches are based on a generative linear model of appearance such as 3D *Morphable Models* [121] or 2D *Active Appearance Models (AAMs)* [25, 71] which have also been extended to 3D [127]. Stemming from Lucas and Kanade's seminal work [68] on image registration, model-based tracking is posed as an optimisation problem minimising a similarity measure between a reference template and the new target image. Successful 3D model-based tracking algorithms based on the low-rank shape basis model include the work of Brand and Bhotika [14] who propose a Bayesian formulation for model-based flexible flow and the efficient approach of Muñoz *et al*. [77, 78] to tracking with 3D *Morphable Models*. All these approaches re-parametrise the image displacements in terms of the model parameters which results in hard constraints.

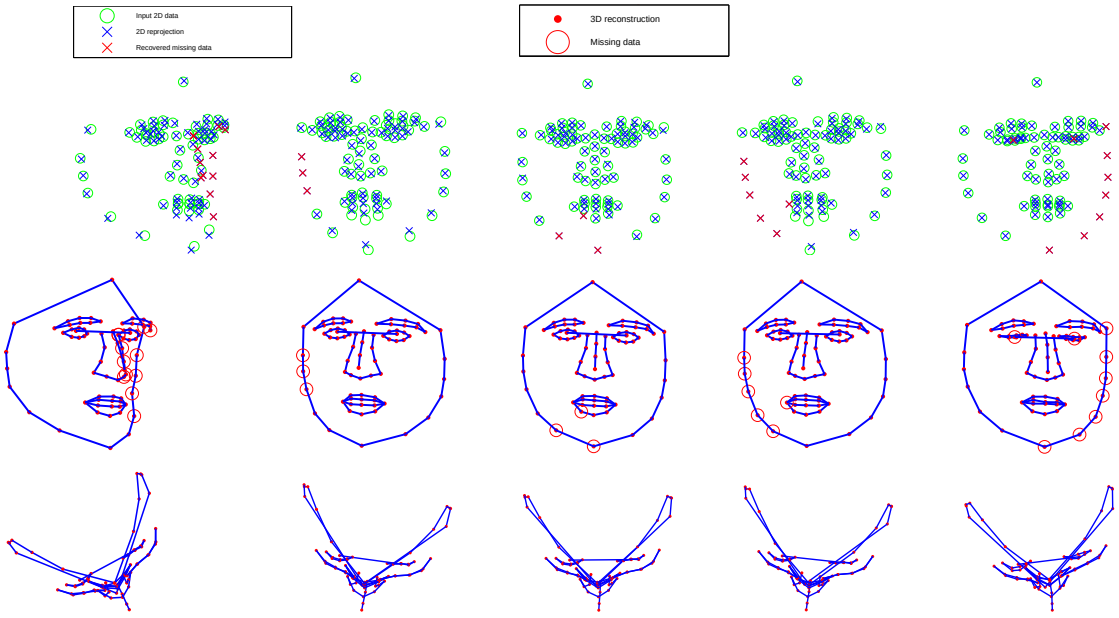


Figure 5.13: Real sign language sequence. Results from the sequential method. First row: input data, 2D reprojection, and recovered missing data. Second and third row: 3D reconstruction front and top views, missing data highlighted with a red circle.

In this section we describe an approach to model-based non-rigid tracking based on soft constraints. We solve simultaneously for the 2D feature tracking (the displacements throughout the sequence of salient points detected in the first frame) and the 3D non-rigid tracking (pose and deformations of the 3D object). We assume a low-rank shape basis has been previously learnt and we formulate tracking as an optimisation problem where our cost function consists of a data term that minimises brightness constancy and a prior term that penalises model parameters (3D object pose and deformations) that deviate from the pre-computed deformable model. Therefore, our model is imposed as a soft rather than a hard constraint. Moreover, we also incorporate spatial and temporal smoothness priors to avoid ambiguities.

5.10.1 Formulation

Two significant difficulties arise in non-rigid tracking. First, the image displacements between consecutive frames are large since we deal with deformable motion. Secondly, as a consequence

of the non-rigidity of the motion, multiple transformations can explain the same pair of images causing ambiguities to arise. In this work, we assume that the tracked feature points lie on a non-rigid 3D surface that deforms according to a known non-rigid low-rank basis and are then projected onto the image via an orthographic camera. While the 3D shape basis will be known in advance, the parameters of the model (camera matrices and deformation parameters) must be estimated at the same time as the image feature displacements. We propose a method for model-based tracking that incorporates the knowledge about the model as a soft constraint. Given a pair of consecutive frames, we seek to estimate the image displacements for each feature point as well as the model parameters that align the projection of the object with the current frame.

The general problem of tracking feature points using the image brightness constancy is to estimate the image displacement vectors δ_p for each feature point p solving the following minimisation problem:

$$\underset{\delta}{\operatorname{argmin}} \sum_{p=1}^{P} ||I(\mathbf{x}_p) - I'(\mathbf{x}_p + \delta_p)||^2 \tag{5.19}$$

where, $\mathbf{x}_p$ is the location of feature p in the reference frame I, δ_p is its displacement in the target frame I' and $I(\mathbf{x}_p)$ indicates the image intensity value at location $\mathbf{x}_p$. The problem of feature tracking is that of estimating the displacements for every feature point that minimise the discrepancy in image intensity between the location of the feature in the reference frame and its location in the target frame. However, the brightness constancy alone cannot provide enough constraints to solve for the image displacements due to the aperture problem. Instead, usually a linear approximation of the brightness constancy equation is performed, assuming a motion model of a patch centred around each feature. We will denote an image patch centred around point $\mathbf{x}$ as the matrix $\mathtt{I}(\mathbf{x})$ containing the interpolated brightness values.

Given a known model, one possible approach to formulate model-based tracking is to re-parametrise the displacement of feature points from one frame to the next in terms of the model parameters. In that case, the cost function is optimised with respect to the model parameters instead of image

displacements:

$$\underset{\mathbf{u}}{\operatorname{argmin}} \sum_{p=1}^{P} ||\mathtt{I}(\mathbf{x}_p) - \mathtt{I}'(\mathbf{f}_p(\mathbf{u}))||_F^2 \tag{5.20}$$

where the model takes the form of a function $\mathbf{f}_p(\cdot)$ that takes as input a vector $\mathbf{u}$ that encodes the current parameters and returns the image feature location for point p. The notation $||\cdot||_F$ indicates the Frobenius norm.

Alternatively, the idea of using a soft constraint is to estimate both feature point displacements δ_p and model parameters $\mathbf{u}$ simultaneously, such that the cost function continues to optimise brightness constancy while penalising displacements that do not satisfy the model. This leads to the alternative cost function:

$$\underset{\delta,\mathbf{u}}{\operatorname{argmin}} \sum_{p=1}^{P} (||\mathtt{I}(\mathbf{x}_p) - \mathtt{I}'(\mathbf{x}_p + \delta_p)||_F^2 + \lambda ||(\mathbf{x}_p + \delta_p) - \mathbf{f}_p(\mathbf{u})||^2) \tag{5.21}$$

We favour imposing the model-based prior as a soft constraint to re-parametrisation of the image displacements for a number of reasons:

- The model is often inaccurate.
- We allow object deformations that are outside of (although close to) the parameter space.
- The cost function has increased robustness to noise.
- The data term for each feature point is independent of the others.

5.10.2 Forward model

While our new *implicit 3D model* described in section 5.3.2 was advantageous for the sequential model-building stage (given its ability to represent deformation modes of any rank), it is not clear that this representation of the low-rank shape model offers an advantage for tracking. In practice, Bregler *et al.*'s *explicit 3D model* has a lower number of time-varying parameters which makes it preferable for model-based tracking.

Fortunately, converting the *implicit 3D model* into its equivalent *explicit 3D model* is simple, re-arranging the elements of the shape matrix and performing a PCA decomposition.

$$\mathtt{S}_{3F\times P} = \begin{bmatrix} \mathtt{S}_1 \\ \mathtt{S}_2 \\ \vdots \\ \mathtt{S}_F \end{bmatrix} = \begin{bmatrix} X_{11} & X_{12} & & X_{1P} \\ Y_{11} & Y_{12} & \cdots & Y_{1P} \\ Z_{11} & Z_{12} & & Z_{1P} \\ \vdots & \vdots & & \vdots \\ X_{F1} & X_{F2} & & X_{FP} \\ Y_{F1} & Y_{F2} & \cdots & Y_{FP} \\ Z_{F1} & Z_{F2} & & Z_{FP} \end{bmatrix} = \begin{bmatrix} \bar{\mathtt{S}} \\ \bar{\mathtt{S}} \\ \vdots \\ \bar{\mathtt{S}} \end{bmatrix} + \begin{bmatrix} \mathtt{U}_1 \\ \mathtt{U}_2 \\ \vdots \\ \mathtt{U}_F \end{bmatrix} \mathtt{V} \tag{5.22}$$

It is straightforward to re-shape the $3F \times P$ matrix of 3D shapes for all frames into a matrix $\mathtt{S}^*$ of size $F \times 3P$, by transposing the coordinates of each 3D point:

$$\mathtt{S}^* = \begin{bmatrix} X_{11} & Y_{11} & Z_{11} & \cdots & X_{1P} & Y_{1P} & Z_{1P} \\ X_{21} & Y_{21} & Z_{21} & \cdots & X_{2P} & Y_{2P} & Z_{2P} \\ \vdots & & & \cdots & & & \vdots \\ X_{F1} & Y_{F1} & Z_{F1} & \cdots & X_{FP} & Y_{FP} & Z_{FP} \end{bmatrix} \tag{5.23}$$

A PCA decomposition of rank K of $\mathtt{S}^*$ gives $\mathtt{LB}^*$, where $\mathtt{L}$ is the $F \times K$ matrix of deformation weights l_{ik}, and the $K \times 3P$ matrix $\mathtt{B}^*$ can be rearranged to give the basis shapes $\mathtt{B}_k$. The *explicit* 3D model therefore only needs K deformation modes to express the shape at each frame, while the *implicit* 3D model needed 3 times as many. This results in fewer parameters to estimate in the tracking stage.

5.10.3 Tracking

Our model-based tracking algorithm is based on the assumption that the 2D points on the image arise from the projection, via an orthographic camera matrix, of 3D points on a non-rigid surface that deforms according to a given *explicit* low-rank basis shape model $\mathtt{B}_d$ to give a matrix of

image measurements for each frame $\mathtt{W}_f$ such that:

$$W_f = \mathtt{R}_f \sum_{d=1}^{K} (l_{fd} \mathtt{B}_d) + \mathtt{T}_f \tag{5.24}$$

Gathering the unknown model parameters ($\mathtt{R}_f$, $\mathbf{l}_f$, $\mathtt{T}_f$) into a parameter vector $\mathbf{u}_f$, the 2D location of a feature point p becomes a real-valued function in this parameter space. The overall parameter vector we wish to optimise in eq. 5.21 contains both the image displacements and the model parameters:

$$\mu_f = [\delta_f, \mathbf{u}_f]^T$$

We adopt the sliding window approach we described in section 5.5 optimising the parameters for ω consecutive frames:

$$\bar{\mu} = [\mu_1, \mu_2, \cdots, \mu_\omega]^T$$

This leads us to formulate the problem as a minimisation over the space of 2D displacements and model parameters for each pair of frames. The overall cost function can be written as:

$$\begin{aligned} \chi(\delta_f, \mathbf{u}_f) = \sum_{p=1}^{P} ||\mathtt{I}_f(x_{f,p}) - \mathtt{I}_{f+1}(x_{f,p} + \delta_p)||_F^2 + \\ + \lambda_1 ||(x_{f,p} + \delta_p) - \mathbf{f}_p(\mathbf{u}_f)||^2 + \\ + \lambda_2 ||u_f - u_{f-1}||^2 \end{aligned} \tag{5.25}$$

The first term of the energy is the data fidelity term which encodes the brightness constancy constraint. It is based on the assumption that the brightness of every feature point p in the one frame is preserved at its new location in the next frame. The second term penalises displacements that do not agree with the model. It gives rise to a soft constraint which is used to enforce the forward model (represented by the $f(\cdot)$ function):

$$f(\mathtt{R}_f, \mathtt{L}_f, \mathbf{t}_f) = \mathtt{W}_f = \mathtt{R}_f \sum_{d} (l_{fd} \mathtt{B}_d) + t_f \tag{5.26}$$

The third term encodes temporal smoothness priors on the parameters, penalising large changes from one frame to the next. We optimise the energy with respect to the model parameters $\mathtt{R}_f, l_{fd}, \mathbf{t}_f$ and the displacements δ_f, using Levenberg Marquardt. Our implementation takes advantage of the sparse nature of the Jacobian matrix which has the form:

$$\mathtt{J} = \begin{pmatrix} \mathtt{D} & \mathtt{0} \\ \mathtt{I} & \mathtt{F} \\ \mathtt{0} & \mathtt{I} \end{pmatrix} \tag{5.27}$$

with a diagonal block $\mathtt{D}$, a full block $\mathtt{F}$ and some zero $\mathtt{0}$ and identity $\mathtt{I}$ blocks. In order to be able to deal with large displacements, we embed our optimisation within a coarse-to-fine approach. We minimise the cost 5.25 over multiple Gaussian pyramid levels, starting at the coarsest level initialising the displacements to zero $\delta = \mathbf{0}$ and using the output of each level to initialise the next.

5.10.4 Experiments

For the purpose of quantitative evaluation of non-rigid model-based tracking we have used a benchmark sequence with ground truth [44]. The sequence uses sparse motion capture (MOCAP) data from [124] to capture the real deformations of a waving flag in 3D. Figure 5.15 shows the 3D motion capture data used to generate a set of rendered images. The 3D surface was then projected synthetically onto the image plane using an orthographic camera and texture mapped to render 450 frames of size 500×500 pixels (see figure 5.14). The advantage of this new sequence is that, since it is based on MOCAP data, it captures the complex natural deformations of a real non-rigid object while allowing us to have access to dense ground truth optical flow.

Figure 5.16 shows the computation of Gaussian pyramids. We use 5 pyramid levels with a down-sampling factor of .5. Figure 5.17 shows results of the tracking obtained using the well-known KLT tracking algorithm [109] to this image sequence. We used the public implementation

Figure 5.14: Some frames of the synthetic flag sequence [44]. 3D and 2D ground truth is available, as the images are generated by texture-mapping known motion capture data of a flag waving in the wind.

provided by Stan Birchfield [3]. Is is clear from the results shown in Figure 5.17 that the simple motion model assumed by the KLT approach is not sufficient to track the feature points due to the large deformations present in the data. By taking advantage of a known 3D model, our method can successfully track feature points, as shown in Figure 5.19 and recover the pose and deformations of the 3D shape as shown in Figure 5.18. In this synthetic experiment the low-rank basis shapes model with $K = 24$ basis shapes is obtained by PCA decomposition of the ground

[3] available at http://www.ces.clemson.edu/~stb/klt/

Figure 5.15: Some frames of the motion capture data used to generate the ground-truth sequence. This synthetic sequence is generated by motion capture data of a flag waving in the wind, performing strong deformations. We use the ground truth 3D data to build our 3D model to perform model-based feature tracking.

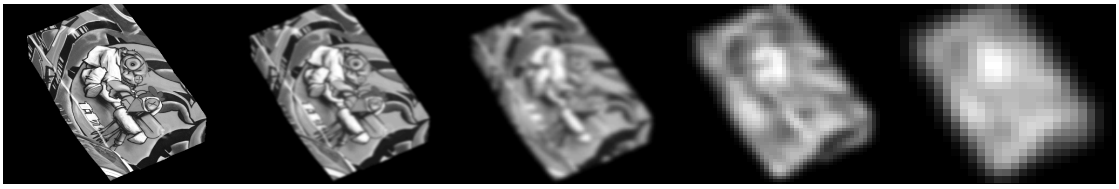


Figure 5.16: Set of Gaussian pyramids: finer to coarser from left to right. Computing Gaussian pyramids allows tracking of a feature point over large displacements.

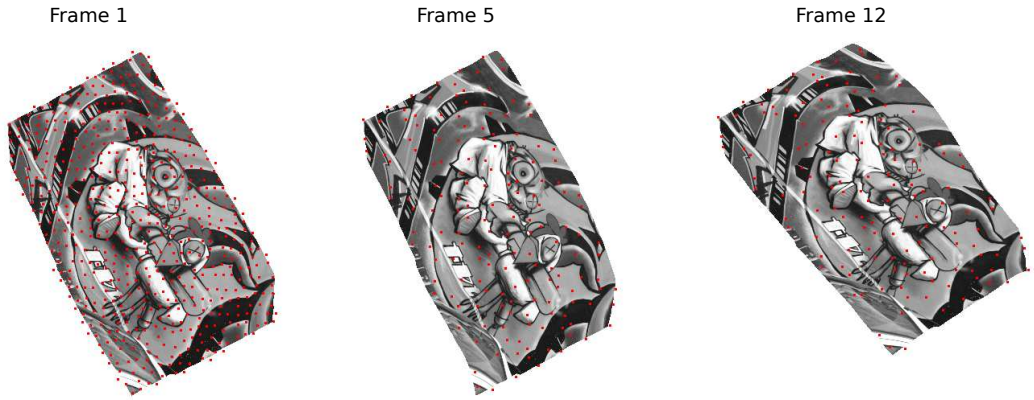
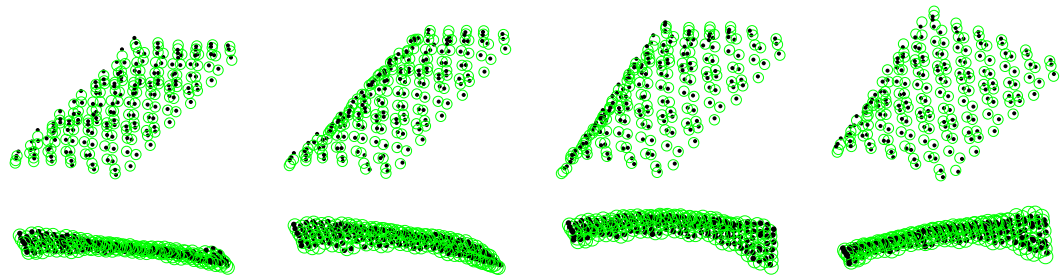


Figure 5.17: Using KLT tracking on the synthetic flag sequence: the strong deformations of the flag cause a great amount of lost tracks. Image brightness is not sufficient to track such deformations.

truth motion capture data (shown in Figure 5.14). We selected 180 equally spaced model points to track and used the ground truth deformation weights as initialisation. The size of the image patch for the brightness constancy was set to 5×5 pixels. The resulting average rms 2D tracking error was 1.5 pixels. The 3D shape is obtained using the computed deformation weights and the known basis shapes. The 3D reconstruction error in this experiment was 4.3%. Some frames of the reconstructed 3D shapes are shown in Figure 5.18.

5.11 Summary and critique

We have undergone a re-thinking of the NRSfM problem for monocular sequences providing a sequential solution. Our new sequential algorithm is able to automatically detect and increase the complexity of the model. Current state of the art methods for NRSfM are batch and rely on prior



Frame 2 Frame 5 Frame 8 Frame 10

Figure 5.18: 3D reconstruction results on the synthetic flag sequence. First row: frontal view. Second row: side view. Reconstructed 3D points shown in black, the ground truth feature points shown as green circles.

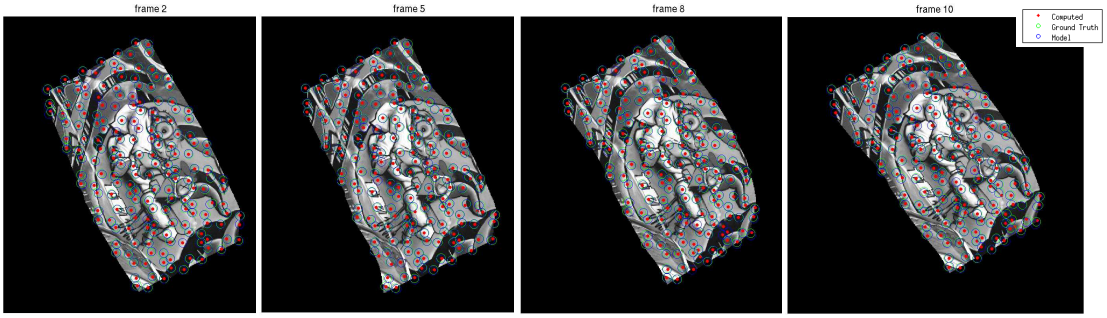


Figure 5.19: Tracking feature points in the synthetic flag sequence: the model keeps the positions constrained, in spite of strong deformations. Feature points computed by our method are shown in red, while the green circles are the known GT locations. The blue circles show the reprojection of features using the computed model parameters.

knowledge of the model complexity (usually the number of basis shapes, K). Our 3D-implicit low-rank shape model simplifies the projection model and allows the rank to grow one-by-one making it well suited to frame-by-frame operation. We have shown quantitative results on a motion capture sequence and shown our system in operation on real sequences. Concerning real time capability, our current MATLAB implementation is not real time (averaging at 1 frame per second in current tests). However, the sliding window approach and the model compaction ensure that the computation time per frame is bounded i.e. does not grow with the number of frames. Therefore we foresee that with appropriate code optimisation we will be able to achieve real-time performance. In addition, in section 5.10.3 we have shown an application to model-

based feature tracking where a model can provide improvements to the feature tracking process using soft constraints. Future work will be directed at combining the sequential frame-by-frame model building with model-based tracking, making our formulation suitable for solving the non-rigid structure from motion problem directly from image streams.

Chapter 6

Conclusions

This thesis addressed the problem of non-rigid structure from motion. We focused on the reconstruction of 3D shapes from a monocular sequence, where neither prior information on the scene nor camera calibration is available. This sfm problem is the most challenging, and has attracted considerable attention in the literature. The ability to solve the general, monocular, uncalibrated case of non-rigid structure from motion is a fundamental task of computer vision, as well as having a wealth of applications in practical domains. This chapter summarises our contributions to the field, and future research directions that are currently open.

6.1 Non-rigid Structure from Motion using Metric Projections

In our work on metric projections in Chapter 3, we show that state of the art results can be obtained by using the manifold constraints of the problem alone. We have used an alternation approach combined with a projection step. Our method obtains the global optimum on the projection problem, that is, each projection is the best point on the manifold of metric constraints, given the current structure estimate. We unified the problems of articulated and deformable structure recovery within a single framework, in which the core of the problem relies solely upon

the manifold projection. In the case of articulated manifold, we propose a convex relaxation to the projection problem.

The problem of non-rigid structure from motion is both inherently ambiguous and non-linear, as the works by Xiao *et al*. and Akther *et al*. have shown. Our proposed algorithm correctly applies non-linear estimation methods for the projection step, obtaining a solution without the use of any additional priors, the process is purely data-driven.

Almost all real-world sequences suffer from missing data and incomplete tracks, and here we show state of the art results. Our experimental results show that projection onto the correct motion manifold makes the method robust to a high percentage of missing data, and encourages viable reconstructions in scenarios where occlusions are not random, but structured, for example due to self occlusions.

We have released the source code for our metric projection method, which quickly become used by other researchers in the field. These researchers gave us valuable new insights, for example, Fayad *et al*. [41] showed a test case in which very strong deformations are not reconstructed by our method, and instead propose a piecewise approach; Taylor *et al*. [107] have tested our method with a sequence containing little or no rotation and translation between the object and the camera, and also suggested to solve this case with a local model. For these challenging cases, local methods clearly are a valuable tool. However, for sequences that can be reconstructed using a low-rank basis shapes model, our method consistently provided state of the art performance.

6.2 Bilinear problems in Computer Vision

The generic optimisation of the metric projection method can be applied to any bi-linear problem with manifold constraints. The BALM algorithm detailed in Chapter 4 provides a general optimisation framework which decouples the problems of factorisation and manifold constraints, and deals with missing data as an additional unknown variable to estimate. It has been applied successfully on computer vision problems outside non-rigid structure from motion, such as photometric stereo, and non-rigid image registration. The BALM method shows fast convergence

thanks to the use of Lagrange multipliers to enforce the metric constraints. This allows the BALM algorithm to solve large-scale problems. One important advantage of the BALM algorithm is its modularity, only a projection onto the manifold defined by the problem constraints is required to apply it to a new bi-linear problem. It is however a global method, and operates in batch, after image acquisition has taken place. For this reason, in Chapter 5 we moved our focus of research to sequential estimation.

6.3 The challenge of real-time estimation

At the time of writing, our sequential estimation method remains the only attempt to solve the non-rigid structure from motion problem on-line, without processing the whole sequence. The key insight in this algorithm which allows us to do this is to decouple the problem of estimating time-varying parameters from that of model building. This two-process approach has already been exploited successfully in rigid camera localisation and mapping.

Our novel 3D-implicit low-rank formulation makes it easy to sequentially increase the rank of the model without recomputing earlier frames. This is particularly desirable in non-rigid structure form motion, when the size of the model (the number of basis shapes K) is unknown. The modelling is guided by the reprojection error. This way of rank-growing allows the automatic detection of new deformation modes. We estimate the new model in two steps, a linear initialisation followed by a non-linear refinement. Our strategy is currently to re-estimate camera and shape parameters when the model is changed. In this case, past frames could be estimated in parallel to the new frames to improve speed.

We demonstrate our method on a MATLAB implementation that currently averages at 1 frame per second, one of the future works will be to rewrite this method with an optimised C++ implementation with the aim of achieving about 15 frames per second, which would make the run-time of non-rigid structure from motion on par with current real-time methods for rigid SfM and SLAM (simultaneous localisation and mapping). We have released code to promote progress in this area.

In order for us to use reprojection error as a measure of fitness for our model, we require reliable tracks. In Chapter 5, we propose the joint estimation of feature points tracks and time-varying model parameters. The applicability of this method was demonstrated in a synthetic sequence.

6.4 Future work

The field of non-rigid structure from motion is maturing, with a wealth of well-understood methods and algorithms. Despite all efforts, no method exists today that can provide a global optimum for both shape and motion estimation, as we have shown, the non-linearities inherent in non-rigid reconstruction make this difficult. We proposed a convex relaxation for the projection on the motion manifold. Further work towards the goal of optimality should take on the problem of robust estimation in presence of high noise and outliers, as well as dealing with missing data. On-line estimation of deformable 3D models raises a number of challenges for future research. First, the sequential frame-by-frame model building needs reliable tracking data. It is possible to improve on the tracking process by using feedback from the model update in the tracking stage. Further research is required to make the feature tracking process robust to outliers, such as features that do not fit the model well. Further, while occlusions result in missing data in a batch processing, in the case of sequential estimation, when some points disappear out of view, other new points appear and can be tracked. Successfully incorporating the new points into an existing model is another direction of research. The new points must be observed reliably for a number of frames before being incorporated in the model. Detection could be guided both by the current model and by a prediction of camera pose according to its current velocity.

Finally, human motion can be seen as a combination of different deformable, articulated, and rigid parts. A hierarchical model building would be a promising approach to recover human motion from a video sequence, by reconstructing the underlying rigid motion as a coarse estimation, the articulated links in a finer level, and a non-rigid motion to capture detailed 3D movements.

Bibliography

[1] H. Aanæs and F. Kahl. Estimation of deformable structure and motion. In *Workshop on Vision and Modelling of Dynamic Scenes, Copenhagen, Denmark*, 2002. 19, 20, 51, 52, 53, 66

[2] S.A. Adeshina and T.F. Cootes. Constructing part-based models for groupwise registration. In *Biomedical Imaging: From Nano to Macro, 2010 IEEE International Symposium on*, April 2010. 69

[3] L. Agapito, E. Hayman, and I. Reid. Self-calibration of rotating and zoooming cameras. *International Journal of Computer Vision*, 45(2):107–127, August 2001. 32

[4] S. Agarwal, N. Snavely, I. Simon, S.M. Seitz, and R. Szeliski. Building rome in a day. In *Computer Vision, 2009 IEEE 12th International Conference on*, October 2009. 33, 34

[5] I. Akhter, Y. Sheikh, and S. Khan. In Defense of Orthonormality Constraints for Nonrigid Structure from Motion. In *IEEE Conference on Computer Vision and Pattern Recognition, Miami, Florida*, Miami, FL, June 2009. 59

[6] I. Akhter, Y. Sheikh, S. Khan, and T. Kanade. Nonrigid Structure from Motion in Trajectory Space. In *Neural Information Processing Systems*, 2008. 75

[7] I. Akhter, Y. Sheikh, S. Khan, and T. Kanade. Trajectory space: A dual representation for nonrigid structure from motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2010. 75, 76, 78

[8] A. Bartoli, V. Gay-Bellile, U. Castellani, J. Peyras, S. Olsen, and P. Sayd. Coarse-to-Fine Low-Rank Structure-from-Motion. In *Proc. IEEE Conference on Computer Vision and*

Pattern Recognition, Anchorage, Alaska, 2008. 19, 20, 52, 53, 67, 71, 94, 101, 105, 154, 162, 172

[9] B. Bascle and A. Blake. Separability of pose and expression in facial tracking and animation. In *Proc. 6th International Conference on Computer Vision, Bombay, India*, 1998. 134

[10] R. Basri, D. Jacobs, and I. Kemelmacher. Photometric stereo with general, unknown lighting. *International Journal of Computer Vision*, 2007. 29, 134

[11] D.P. Bertsekas. *Constrained optimization and Lagrange multiplier methods*. Academic Press New York, 1982. 139

[12] M. Brand. Morphable models from video. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, Kauai, Hawaii*, December 2001. 19, 52, 53, 60, 71

[13] M. Brand. A direct method for 3D factorization of nonrigid motion observed in 2D. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, San Diego, California*, 2005. 56, 59, 60

[14] M. Brand and R. Bhotika. Flexible flow for 3d nonrigid tracking and shape recovery. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, Kauai, Hawaii*, pages 315–22, December 2001. 174

[15] C. Bregler, A. Hertzmann, and H. Biermann. Recovering non-rigid 3D shape from image streams. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, Hilton Head, South Carolina*, June 2000. 18, 19, 44, 46, 47, 49, 50, 52, 53, 56, 71, 75, 90, 95, 134, 142, 156, 157, 159

[16] F. Brunet, R. Hartley, A. Bartoli, N. Navab, and R. Malgouyres. Monocular template-based reconstruction of smooth and inextensible surfaces. In *Proc. 10th Asian Conference on Computer Vision, Queenstown, New Zealand*, 2010. 51, 78, 80

[17] A. M. Buchanan and A. Fitzgibbon. Damped newton algorithms for matrix factorization with missing data. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, San Diego, California*, 2005. 71, 99, 109, 135, 148

[18] E. Candès and B. Recht. Exact matrix completion via convex optimization. *Foundations of Computational Mathematics*, 2009. 71

[19] M. Chandraker and D. Kriegman. Globally optimal bilinear programming for computer vision applications. In *IEEE Conference on Computer Vision and Pattern Recognition, 2008. CVPR 2008*, 2008. 136

[20] P. Chen. Optimization algorithms on subspaces: Revisiting missing data problem in low-rank matrix. *International Journal of Computer Vision*, 2008. 71

[21] P. Chen and D. Suter. Recovering the missing components in a large noisy low-rank matrix: application to sfm. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, August 2004. 70

[22] K.M.G. Cheung, S. Baker, and T. Kanade. Shape-from-silhouette of articulated objects and its use for human body kinematics estimation and motion capture. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, Madison, Wisconsin*, June 2003. 30

[23] A.R Conn, N. Gould, A. Sartenaer, and P.L. Toint. Convergence properties of an augmented Lagrangian algorithm for optimization with a combination of general equality and linear constraints. *SIAM Journal on Optimization*, 1996. 142

[24] T. F. Cootes and C. J. Taylor. Active shape models – smart snakes. In *Proc. British Machine Vision Conference*, pages 266–275, 1992. 67

[25] T.F. Cootes, G.J. Edwards, and C.J. Taylor. Active appearance models. In *Proc. 6th European Conference on Computer Vision, Dublin, Ireland*, 1998. 67, 174

[26] T.F. Cootes, C.J. Twining, V.S. Petrović, K.O. Babalola, and C.J. Taylor. Computing accurate correspondences across groups of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2010. 69

[27] T.F. Cootes, C.J. Twining, V.S. Petrović, R. Schestowitz, and C.J. Taylor. Groupwise construction of appearance models using piece-wise affine deformations. In *Proc. 16th British Machine Vision Conference, Oxford*, 2005. 69

[28] J. Costeira and T. Kanade. A multibody factorization method for independent moving objects. *International Journal of Computer Vision*, September 1998. 17, 42, 81

[29] Y. Dai, H. Li, and M. He. Element-wise factorization for n-view projective reconstruction. In *Proc. 11th European Conference on Computer Vision, Crete, Greece*, 2010. 44

[30] R.H. Davies, C.J. Twining, T.F. Cootes, J.C. Waterton, and C.J. Taylor. A minimum description length approach to statistical shape modeling. *Medical Imaging, IEEE Transactions on*, May 2002. 69

[31] A. Del Bue. *Deformable 3-D Modelling from Uncalibrated Video Sequences*. PhD thesis, Queen Mary University of London, August 2006. 66

[32] A. Del Bue. A factorization approach to structure from motion with shape priors. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, june 2008. 52, 67

[33] A. Del Bue and L. Agapito. Non-rigid 3D shape recovery using stereo factorization. *Asian Conference of Computer Vision*, January 2004. 67

[34] A. Del Bue and L. Agapito. Stereo non-rigid factorization. *International Journal of Computer Vision*, February 2006. 66

[35] A. Del Bue, X. Lladó, and L. Agapito. Non-rigid metric shape and motion recovery from

uncalibrated images using priors. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, New York, NY*, 2006. 19, 20, 51, 52, 53, 71

[36] A. Del Bue, F. Smeraldi, and L. Agapito. Non-rigid structure from motion using ranklet–based tracking and non-linear optimization. *Image and Vision Computing*, March 2007. 19, 20, 52, 101, 111, 113, 131, 146

[37] A. Del Bue, J. Xavier, L. Agapito, and M. Paladini. Bilinear factorization via augmented lagrange multipliers. In *Proc. 11th European Conference on Computer Vision, Crete, Greece*. Springer, 2010. 141, 144

[38] M. Dodig, M. Stošić, and J. Xavier. On minimizing a quadratic function on stiefel manifolds. Technical report, Instituto de Sistemas e Robotica, 2009. Available at http://users.isr.ist.utl.pt/˜jxavier/ctech.pdf. 102, 104, 206

[39] A. Edelman, T.A. Arias, and S.T. Smith. The geometry of algorithms with orthogonality constraints. *SIAM Journal on Matrix Analysis and Applications*, 1998. 104, 105

[40] O. D. Faugeras, Q. Luong, and S. Maybank. Camera self-calibration: Theory and experiments. In *Proc. European Conference on Computer Vision*, LNCS 588, pages 321–334, 1992. 17

[41] J. Fayad, L. Agapito, and A. Del Bue. Piecewise quadratic reconstruction of non-rigid surfaces from monocular sequences. In *Proc. 11th European Conference on Computer Vision, Crete, Greece*, 2010. 51, 52, 64, 73, 75, 186

[42] J. Fayad, A. Del Bue, L. Agapito, and P. Aguiar. Non-rigid structure from motion using quadratic deformation models. In *British Machine Vision Conference, London, UK*, 2009. 73

[43] J. Fayad, C. Russell, and L. Agapito. Automated articulated structure and 3d shape re-

covery from point correspondences. *13th International Conference on Computer Vision, Barcelona, Spain*, November 2011. 74

[44] R. Garg, A. Roussos, and L. Agapito. Robust trajectory-space tv-l1 optical flow for non-rigid sequences. In *Energy Minimazation Methods in Computer Vision and Pattern Recognition - 8th International Conference, EMMCVPR 2011, St. Petersburg, Russia*, July 2011. 180, 181

[45] P.F.U. Gotardo and A.M. Martinez. Computing smooth time-trajectories for camera and deformable shape in structure from motion with occlusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2011. 78, 173

[46] R. Guerreiro and P. Aguiar. Estimation of rank deficient matrices from partial observations: Two-step iterative algorithms. In *Energy Minimization Methods in Computer Vision and Pattern Recognition*, 2003. 70

[47] R. Hartley and F. Schaffalitzky. PowerFactorization: 3D reconstruction with missing or uncertain data. In *Proc. Australia-Japan Advanced Workshop on Computer Vision*, 2003. 70

[48] R. Hartley and F. Schaffalitzky. Reconstruction from projections using grassmann tensors. In *Proc. 6th European Conference on Computer Vision, Dublin, Ireland*, 2004. 60

[49] R. Hartley and R. Vidal. Perspective nonrigid shape and motion recovery. In *Proc. European Conference on Computer Vision*, 2008. 56, 60, 98

[50] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition, 2004. 16, 31

[51] R. I. Hartley. Self-calibration from multiple views with a rotating camera. In *Proc. 3rd European Conference on Computer Vision, Stockholm*, volume 1, pages 471–478, 1994. 32

[52] R. I. Hartley. Self-calibration of stationary cameras. *International Journal of Computer Vision*, 22(1):5–23, February 1997. 32

[53] C. Hernández, G. Vogiatzis, G. J. Brostow, B. Stenger, and R. Cipolla. Non-rigid photometric stereo with colored lights. In *Proc. of the 11th IEEE International Conference on Computer Vision (ICCV)*, 2007. 29

[54] M.R. Hestenes. Multiplier and gradient methods. *Journal of Optimization Theory and Applications*, 1969. 139

[55] A. Heyden. Projective structure and motion from image sequences using subspace methods. In *Scandinavian Conference on Image Analysis*, June 1997. 44

[56] A. Heyden and K. Åström. Flexible calibration: Minimal cases for auto-calibration. In *Proc. 7th International Conference on Computer Vision, Kerkyra, Greece*, pages 350–355, 1999. 32

[57] B. K.P. Horn. Shape from shading: A method for obtaining the shape of a smooth opaque object from one view. Technical report, Massachusetts Institute of Technology, Cambridge, MA, USA, 1970. 29

[58] D. Jacobs. Linear fitting with missing data for structure-from-motion. *Computer Vision and Image Understanding*, 2001. 70

[59] K. Kanatani and Y. Sugaya. Factorization without factorization: complete recipe. *Memories of the Faculty of Engineering, Okayama University*, 2004. 42

[60] G. Klein and D. Murray. Parallel tracking and mapping for small AR workspaces. In *Proc. Sixth IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR'07)*, Nara, Japan, November 2007. 34, 155, 167

[61] K. N. Kutulakos and S. M. Seitz. A theory of shape by space carving. *International Journal of Computer Vision*, 2000. 30

[62] A. Laurentini. The visual hull of curved objects. In *Proc. 7th International Conference on Computer Vision, Kerkyra, Greece*, 1999. 29

[63] Z. Lin, M. Chen, L. Wu, and Y. Ma. The Augmented Lagrange Multiplier Method for Exact Recovery of Corrupted Low-Rank Matrices. *UIUC Technical Report UILU-ENG-09-2215*, 2009. 136

[64] X. Lladó, A. Del Bue, and L. Agapito. Euclidean reconstruction of deformable structure using a perspective camera with varying intrinsic parameters. In *Proc. International Conference on Pattern Recognition*, Hong Kong, 2006. 67

[65] H. C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, September 1981. 17

[66] M.I. A. Lourakis and A.A. Argyros. SBA: A Software Package for Generic Sparse Bundle Adjustment. *ACM Transactions on Mathematical Software (TOMS)*, 2009. 33

[67] D. Lowe. Distinctive image features from scale-invariant keypoints. In *International Journal of Computer Vision*, 2003. 31

[68] B.D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proc. International Joint Conference on Artificial Intelligence*, 1981. 40, 174

[69] F. Mai and Y. S. Hung. Augmented lagrangian-based algorithm for projective reconstruction from multiple views with minimization of 2d reprojection error. *Journal of Signal Processing Systems*, 2009. 136

[70] M. Marques and J. Costeira. Estimating 3d shape from degenerate sequences with missing data. *Computer Vision and Image Understanding*, 2009. 92, 93, 94, 99, 100, 103, 110, 114, 119, 130, 135, 147, 148, 149

[71] I. Matthews and S. Baker. Active appearance models revisited. *International Journal of Computer Vision*, November 2004. 67, 174

[72] I. Matthews, J. Xiao, and S. Baker. On the dimensionality of deformable face models. Technical Report CMU-RI-TR-06-12, Robotics Institute, Pittsburgh, PA, March 2006. 68

[73] S. Maybank and O. D. Faugeras. A theory of self-calibration of a moving camera. *Int. J. Comput. Vision*, 1992. 32

[74] T. P. Minka. Automatic choice of dimensionality for pca. Technical report, M.I.T. Media Laboratories, 2000. 66

[75] T. Moons, L. Van Gool, M. van Diest, and A. Oosterlinck. Affine structure from perspective image pairs under relative translations between object and camera. Technical Report KUL/ESAT/M12/9306, Departement Elektrotechniek, Katholieke Universiteit Leuven, Belgium, 1993. 32

[76] E. Mouragnon, M. Lhuillier, M. Dhome, F. Dekeyser, and P. Sayd. Generic and real-time structure from motion using local bundle adjustment. *Image and Vision Computing*, 2009. 155

[77] E. Muñoz, J. M. Buenaposada, and L. Baumela. Efficient model-based 3d tracking of deformable objects. In *Proc. International Conference on Computer Vision*, Beijing, China, October 2005. 174

[78] E. Muñoz, J. M. Buenaposada, and L. Baumela. A direct approach for efficiently tracking with 3d morphable models. In *Proc. International Conference on Computer Vision*, 2009. 174

[79] R.A. Newcombe and A.J. Davison. Live dense reconstruction with a single moving camera. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, June 2010. 34

[80] R.A. Newcombe, S. Lovegrove, and A.J. Davison. DTAM: Dense Tracking and Mapping in Real-Time. In *13th International Conference on Computer Vision (ICCV2011)*, November 2011. 34

[81] D. Nister. An efficient solution to the five-point relative pose problem. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, June 2004. 33

[82] S. Olsen and A. Bartoli. Using priors for improving generalization in non-rigid structure-from-motion. *Proc. British Machine Vision Conference*, 2007. 70

[83] S. Olsen and A. Bartoli. Implicit non-rigid structure-from-motion with priors. *Journal of Mathematical Imaging and Vision*, 2008. 157, 159

[84] M. Paladini, A. Bartoli, and L. Agapito. Sequential non-rigid structure-from-motion with the 3d-implicit low-rank shape model. In *Proc. 11th European Conference on Computer Vision, Crete, Greece*, 2010. 52, 173

[85] M. Paladini, A. Del Bue, M. Stosic, M. Dodig, J. Xavier, and L. Agapito. Factorization for non-rigid and articulated structure using metric projections. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2009.*, June 2009. 94, 108, 135, 146, 154

[86] M. Paladini, A. Del Bue, J. Xavier, A. Lourdes, M. Stošić, and M. Dodig. Optimal metric projections for deformable and articulated structure-from-motion. *International Journal of Computer Vision*, 2011. 64, 65, 94

[87] Q. Pan, G. Reitmayr, and T. Drummond. ProFORMA: Probabilistic Feature-based On-line Rapid Model Acquisition. In *Proc. 20th British Machine Vision Conference (BMVC)*, London, September 2009. 34

[88] H. Park, T. Shiratori, I. Matthews, and Y. Sheikh. 3d reconstruction of a moving point

from a series of 2d projections. In *Proc. 11th European Conference on Computer Vision, Crete, Greece*, 2010. 77

[89] M. Perriollat and A. Bartoli. A quasi-minimal model for paper-like surfaces. *Proceedings of the ISPRS International Workshop "Towards Benmarking Automated Calibration, Orientation, and Surface Reconstruction from Images" at CVPR'07, Minneapolis, USA*, June 2007. 51

[90] M. Perriollat, R. Hartley, and A. Bartoli. Monocular template-based reconstruction of inextensible surfaces. *International Journal of Computer Vision*, 2010. 51, 78, 80

[91] M. Pollefeys, R. Koch, and L. Van Gool. Self calibration and metric reconstruction in spite of varying and unknown internal camera parameters. In *Proc. 6th International Conference on Computer Vision, Bombay, India*, 1998. 32, 33

[92] M. Pollefeys and L. Van Gool. Visual modelling: from images to images. *The Journal of Visualization and Computer Animation*, 2002. 31, 32, 34

[93] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes in C*. Cambridge University Press, 1988. 65

[94] V. Rabaud and S. Belongie. Re-thinking non-rigid structure from motion. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, Alaska*, Anchorage, AL, 2008. 52, 74

[95] C. Russell, J. Fayad, and L. Agapito. Energy based multiple model fitting for non-rigid structure from motion. In *IEEE Conference on Computer Vision and Pattern Recognition, Colorado Springs*, June 2011. 51, 52, 64, 74, 75

[96] M. Salzmann and P. Fua. Reconstructing sharply folding surfaces: A convex formulation. In *IEEE Conference on Computer Vision and Pattern Recognition, Miami, Florida*, June 2009. 80

[97] M. Salzmann, R. Hartley, and P. Fua. Convex optimization for deformable surface 3-d tracking. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis, Minnesota, USA*, October 2007. 51, 78, 79

[98] M. Salzmann, F. Moreno-Noguer, V. Lepetit, and P. Fua. Closed-Form Solution to Non-Rigid 3D Surface Registration. In *Proceedings of the European Conference on Computer Vision*, 2008. 78, 80

[99] M. Salzmann, R. Urtasun, and P. Fua. Local deformation models for monocular 3d shape recovery. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, Alaska*, June 2008. 68

[100] A. Shaji, S. Chandran, and D. Suter. Manifold Optimisation for Motion Factorisation. In *19th International Conference on Pattern Recognition (ICPR 2008)*, 2008. 135

[101] J. Shi and C. Tomasi. Good features to track. *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 1994. 31

[102] N. Snavely, S.M. Seitz, and R. Szeliski. Photo tourism: Exploring photo collections in 3d. In *SIGGRAPH Conference Proceedings*, 2006. 33

[103] J.F. Sturm. Using SeDuMi 1.02, A Matlab toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 1999. 102, 104, 108, 206

[104] P. Sturm and B. Triggs. A factorization based algorithm for multi-image projective structure and motion. In *Proc. 4th European Conference on Computer Vision, Cambridge*. Springer-Verlag, 1996. 17, 42, 43, 44

[105] J.K. Tan and S. Ishikawa. Deformable shape recovery by factorization based on a spatiotemporal measurement matrix. *Computer Vision and Image Understanding*, 2001. 52

[106] J.P. Tardif, A. Bartoli, M. Trudeau, N. Guilbert, and S. Roy. Algorithms for batch matrix

factorization with application to structure-from-motion. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis, Minnesota, USA*, 2007. 70

[107] J. Taylor, A.D. Jepson, and K.N. Kutulakos. Non-rigid structure from locally-rigid motion. In *IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, CA*, June 2010. 51, 64, 72, 75, 186

[108] S. Thrun. Affine structure from sound. *Advances in Neural Information Processing Systems*, 2006. 134

[109] C. Tomasi and T. Kanade. Shape and motion from image streams: a factorization method - part 3 detection and tracking of point features. Technical Report CMU-CS-91-132, Computer Science Department, Carnegie Mellon University, Pittsburgh, PA, April 1991. 34, 35, 40, 180

[110] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: A factorization method. *International Journal of Computer Vision*, November 1992. 17, 36, 37, 42, 49, 90, 98, 113, 134, 170, 172

[111] L. Torresani, A. Hertzmann, and C. Bregler. Non-rigid structure-from-motion: Estimating shape and motion with hierarchical priors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2008. 19, 20, 52, 53, 56, 63, 64, 71, 76, 94, 101, 105, 111, 112, 113, 120, 122, 146, 147, 154, 170, 171, 172

[112] L. Torresani, D. Yang, E. Alexander, and C. Bregler. Tracking and modeling non-rigid objects with rank constraints. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, Kauai, Hawaii*, 2001. 19, 20, 52, 53, 62, 64, 70, 91, 92, 93, 100, 110, 131

[113] P. Tresadern and I. Reid. Articulated structure from motion by factorization. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, San Diego, California*, June 2005. 81, 82, 84, 86, 91, 96, 106, 109, 111, 128, 129, 150

[114] B. Triggs. Factorization methods for projective structure and motion. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, San Francisco*, 1996. 44

[115] B. Triggs. Auto-calibration and the absolute quadric. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, Puerto Rico*, pages 609–614, 1997. 32

[116] B. Triggs, P. McLauchlan, R. I. Hartley, and A. Fitzgibbon. Bundle adjustment – A modern synthesis. In *Vision Algorithms: Theory and Practice*. Springer Verlag, 2000. 33

[117] T. Ueshiba and F. Tomita. A factorization method for projective and euclidean reconstruction from multiple perspective views via iterative depth estimation. In *Proc. 5th European Conference on Computer Vision, Freiburg, Germany*, 1998. 44

[118] S. Ullman. *The Interpretation of Visual Motion*. MIT Press, 1979. 31

[119] R. Urtasun, D. Fleet, A. Hertzmann, and P. Fua. Priors for people tracking from small training sets. In *Tenth IEEE International Conference on In Computer Vision.*, 2005. 68

[120] A. Varol, M. Salzmann, E. Tola, and P. Fua. Template-free monocular reconstruction of deformable surfaces. In *Computer Vision, 2009 IEEE 12th International Conference on*, October 2009. 51, 72, 73

[121] T. Vetter and V. Blanz. A morphable model for the synthesis of 3d faces. In *Proceedings of the ACM SIGGRAPH Conference on Computer Graphics*, 1999. 68, 174

[122] R. Vidal and R. Hartley. Motion segmentation with missing data using powerfactorization and gpca. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, Washington D.C.*, 2004. 81

[123] G. Wang, H.T. Tsui, and Q.M.J. Wu. Rotation constrained power factorization for structure from motion of nonrigid objects. *Pattern Recognition Letters*, 2008. 64, 65, 91, 92, 93, 100, 110, 111, 112, 113, 114, 115, 120, 122, 131, 146

[124] R. White, K. Crane, and D. Forsyth. Capturing and animating occluded cloth. In *ACM Transactions on Graphics*, 2007. 180

[125] T. Wiberg. Computation of principal components when data are missing. In *Proc. of the 2nd Symposium on Computational Statistics, Berlin*, 1976. 70, 71, 135

[126] R. J. Woodham. Photometric method for determining surface orientation. *Optical Engineering*, 1980. 29

[127] J. Xiao, S. Baker, I. Matthews, and T. Kanade. Real-time combined 2d+3d active appearance models. In *cvpr2004*, June 2004. 174

[128] J. Xiao, J. Chai, and T. Kanade. A closed-form solution to non-rigid shape and motion recovery. *International Journal of Computer Vision*, April 2006. 19, 56, 57, 59, 60, 71, 98, 157

[129] J. Xiao and T. Kanade. Non-rigid shape and motion recovery: Degenerate deformations. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, Washington D.C.*, 2004. 160

[130] J. Xiao and T. Kanade. Uncalibrated perspective reconstruction of deformable structures. In *Proc. 10th International Conference on Computer Vision, Beijing, China*, October 2005. 56, 98

[131] J. Yan and M. Pollefeys. A factorization-based approach to articulated motion recovery. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, San Diego, California*, June 2005. 81, 82, 83, 106

[132] J. Yan and M. Pollefeys. A general framework for motion segmentation: Independent, articulated, rigid, non-rigid, degenerate and non-degenerate. In *Proc. 4th European Conference on Computer Vision, Cambridge*, 2006. 81

[133] J. Yan and M. Pollefeys. A factorization-based approach for articulated non-rigid shape, motion and kinematic chain recovery from video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, May 2008. 82, 91, 96, 111

[134] C. Zach, T. Pock, and H. Bischof. A duality based approach for realtime tv-l1 optical flow. In *Proceedings of the 29th DAGM conference on Pattern recognition*, 2007. 31

[135] R. Zhang, P. Tsai, J.E. Cryer, and M. Shah. Shape-from-shading: a survey. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, August 1999. 29

[136] S. Zhu, L. Zhang, and B.M. Smith. Model evolution: An incremental approach to non-rigid structure from motion. *IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, CA*, 2010. 75

Appendix A

Optimization, deformable case

For $\mathtt{E} \in \mathbb{R}^{6\times 6}$, our aim is to compute

$$\min_{\mathbf{q}=vec(\mathtt{Q})} \mathbf{q}^T \mathtt{E}\mathbf{q}, \tag{A.1}$$

where $Q \in \mathbb{R}^{3\times 2}$ runs through Stiefel matrices, i.e. $\mathtt{Q}^T\mathtt{Q} = \mathtt{I}_{2\times 2}$. We rewrite (A.1) as

$$\min_{\mathbf{q}=vec(\mathtt{Q})} \mathrm{Tr}(\mathtt{E}\mathbf{q}\mathbf{q}^T) = \min_{\mathtt{X}\in S} \mathrm{Tr}(\mathtt{E}\mathtt{X}), \tag{A.2}$$

where S is the set of all real symmetric 6×6 matrices $\mathtt{X} = \begin{bmatrix} \mathtt{A} & \mathtt{B} \\ \mathtt{B}^T & \mathtt{C} \end{bmatrix}$, with $\mathtt{A} \in \mathbb{R}^{3\times 3}$, satisfying

$$\mathtt{X} \succcurlyeq 0, \tag{A.3}$$

$$\mathrm{Tr}(\mathtt{A}) = \mathrm{Tr}(\mathtt{C}) = 1, \quad \mathrm{Tr}(\mathtt{B}) = 0, \tag{A.4}$$

$$\mathrm{rank}\,\mathtt{X} = 1. \tag{A.5}$$

This problem, has a nonconvex constraint ($\text{rank}\,\mathtt{X} = 1$). Since the cost function is linear we have

$$\min_{\mathtt{X}\in S} \text{Tr}(\mathtt{EX}) = \min_{\mathtt{X}\in \text{co}(S)} \text{Tr}(\mathtt{EX}), \tag{A.6}$$

where $\text{co}(S)$ is the convex hull of the set S. Here, we compute the convex hull (tight convex relaxation) $\text{co}(S)$ as all the real symmetric 6×6 matrices $\mathtt{X}$ that satisfy

$$\mathtt{X} \succcurlyeq 0, \tag{A.7}$$

$$\text{Tr}(\mathtt{A}) = \text{Tr}(\mathtt{C}) = 1, \quad \text{Tr}(\mathtt{B}) = 0, \tag{A.8}$$

$$\begin{bmatrix} \mathtt{I}_{3\times 3} - \mathtt{A} - \mathtt{C} & \mathbf{w} \\ \mathbf{w}^T & 1 \end{bmatrix} \succcurlyeq 0, \tag{A.9}$$

with $\mathbf{w}$ given by

$$\mathbf{w} = \begin{bmatrix} b_{23} - b_{32} \\ b_{31} - b_{13} \\ b_{12} - b_{21} \end{bmatrix} \tag{A.10}$$

where $\mathtt{B} = [b_{ij}]$. Moreover, this set is defined only by linear matrix inequalities (LMI). Hence, we have that our problem (A.1) is equivalent to finding the minimum of a linear function ($\text{Tr}(\mathtt{EX})$) on a convex set ($\text{co}(S)$), which is given only by LMI (A.7)-(A.9). Thus, the optimization problem in the right-hand side of (23) is a Semi-Definite Program (SDP). By using SeDuMi [103], we quickly obtain the optimal matrix $\mathtt{X}$ for (A.6). In 100% of experiments that we ran, the optimal matrix X was always of rank 1. By factorizing $\mathtt{X} = \mathbf{q}\mathbf{q}^T$, we obtain the optimal *Stiefel matrix* as $\mathtt{Q} = vec^{-1}(\mathbf{q})$. For more details the reader can refer to [38]

Appendix B

Convex relaxation, Articulated Case

Problem statement

We consider the optimization problem

$$\begin{array}{ll} \text{maximize} & f(u) \qquad . \\ \text{subject to} & \|u\| \leq 1 \end{array} \tag{B.1}$$

where the variable to optimize is $u \in \mathbb{R}^2$. The objective function is

$$f(u) = \|u\|^2 + 2u^\top x + 2\left\|\left(I - uu^\top\right)^{1/2} Y\right\|_{\mathsf{N}} + 2\left\|\left(I - uu^\top\right)^{1/2} Z\right\|_{\mathsf{N}} \tag{B.2}$$

The problem data is the triple

$$(x, Y, Z) \in \mathbb{R}^2 \times \mathbb{R}^{2\times 2} \times \mathbb{R}^{2\times 2}.$$

For an $n \times n$ matrix X, the symbol $\|X\|_{\mathsf{N}} = \sigma_1(X) + \cdots + \sigma_n(X)$ denotes its nuclear norm.

Problem reformulation

We start by noting that (B.1) is equivalent to maximizing

$$g(u) = \|u\|^2 + 2|u^\top x| + 2\left\|\left(I - uu^\top\right)^{1/2} Y\right\|_{\mathsf{N}} + 2\left\|\left(I - uu^\top\right)^{1/2} Z\right\|_{\mathsf{N}}. \tag{B.3}$$

Note that $f(u) \leq g(u)$ for all feasible u. However, at a global maximizer of (B.1), say $u^\star$, we must have $(u^\star)^\top x \geq 0$. Thus, $(u^\star)^\top x = |(u^\star)^\top x|$ and $f(u^\star) = g(u^\star)$.

We rewrite $g(u)$ as

$$g(u) = \|u\|^2 + 2\sqrt{u^\top x x^\top u} + 2\left\|\left(I - uu^\top\right)^{1/2} Y\right\|_{\mathsf{N}} + 2\left\|\left(I - uu^\top\right)^{1/2} Z\right\|_{\mathsf{N}}. \tag{B.4}$$

Moreover, for a 2×2 matrix X, there holds

$$\|X\|_{\mathsf{N}} = \sqrt{\|X\|^2 + 2|\det(X)|} \tag{B.5}$$

where $\|X\| = \sqrt{\mathrm{tr}\,(XX^\top)}$ denotes the Frobenius norm of X. Using (B.5) in (B.4) gives

$$\begin{aligned} g(u) \;=\; & \|u\|^2 + 2\sqrt{u^\top x x^\top u} + 2\sqrt{\|Y\|^2 - u^\top Y Y^\top u + 2|\det(Y)|\sqrt{1 - u^\top u}} + \\ & + 2\sqrt{\|Z\|^2 - u^\top Z Z^\top u + 2|\det(Z)|\sqrt{1 - u^\top u}}. \end{aligned} \tag{B.6}$$

Now, we distinguish two cases:

1. The matrices $\{I_2, YY^\top, ZZ^\top\}$ are linearly independent

2. The matrices $\{I_2, YY^\top, ZZ^\top\}$ are linearly dependent

Case 1 is probably the one occurring the most in practice. It will lead do a semidefinite program (SDP). Case 2 is easier. It will lead to a 2nd order cone program (SOCP).

Case 1: $\{I_2, YY^\top, ZZ^\top\}$ *are linearly independent*

In this case, the matrices $\{I_2, YY^\top, ZZ^\top\}$ form a basis for the three-dimensional vector space of 2×2 matrices. This means that there exists $\alpha, \beta, \gamma \in \mathbb{R}$ such that

$$xx^\top = \alpha I_2 + \beta YY^\top + \gamma ZZ^\top. \tag{B.7}$$

Using (B.7) in (B.6) yields

$$\begin{aligned} g(u) &= \|u\|^2 + 2\sqrt{\alpha u^\top u + \beta u^\top YY^\top u + \gamma u^\top ZZ^\top u} + \\ &\quad + 2\sqrt{\|Y\|^2 - u^\top YY^\top u + 2|\det(Y)|\sqrt{1-u^\top u}} + \\ &\quad + 2\sqrt{\|Z\|^2 - u^\top ZZ^\top u + 2|\det(Z)|\sqrt{1-u^\top u}}. \end{aligned} \tag{B.8}$$

Our optimization problem is

$$\begin{aligned} &\text{maximize} && g(u) \\ &\text{subject to} && \|u\| \le 1 \end{aligned} \tag{B.9}$$

with $g(u)$ as in (B.8). In (B.9), the variable to optimize is $u \in \mathbb{R}^2$. Problem (B.9) can be rewritten as

$$\begin{aligned} &\text{maximize} && \phi(a,b,c) \\ &\text{subject to} && (a,b,c) \in \mathcal{S} \\ & && a \le 1 \end{aligned} \tag{B.10}$$

where

$$\mathcal{S} := \{(a,b,c) : \exists_u : a = u^\top u,\, b = u^\top YY^\top u,\, c = u^\top ZZ^\top u\},$$

and

$$\begin{aligned} \phi(a,b,c) &:= a + 2\sqrt{\alpha a + \beta b + \gamma c} + 2\sqrt{\|Y\|^2 - b + 2|\det(Y)|\sqrt{1-a}} + \\ &\quad + 2\sqrt{\|Z\|^2 - c + 2|\det(Z)|\sqrt{1-a}} \end{aligned}$$

is a concave function.

We have the inclusion $\mathcal{S} \subset \mathcal{T}$ where

$$\mathcal{T} := \{(a,b,c) : \exists_{U \succeq 0} : a = \mathsf{tr}(U), b = \mathsf{tr}\left(YY^\top U\right), c = \mathsf{tr}\left(ZZ^\top U\right)\}.$$

Using $\mathcal{T}$ instead of $\mathcal{S}$ in (B.10) gives the convex problem

$$\begin{array}{ll} \text{maximize} & \phi(a,b,c) \\ \text{subject to} & a = \mathsf{tr}(U) \\ & b = \mathsf{tr}\left(YY^\top U\right) \\ & c = \mathsf{tr}(ZZ^\top U) \\ & U \succeq 0 \\ & a \leq 1 \end{array} \quad . \tag{B.11}$$

Let $U^\star$ be a solution of (B.11). Let

$$U^\star = \begin{bmatrix} u_1 & u_2 \end{bmatrix} \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \begin{bmatrix} u_1^\top \\ u_2^\top \end{bmatrix}$$

be an eigenvalue decomposition, where $\lambda_1 \geq \lambda_2$. A suboptimal solution for (B.1) is $u^\star = \pm\sqrt{\lambda_1}u_1$, where the sign is chosen such that $x^\top u^\star \geq 0$.

Case 2: $\{I_2, YY^\top, ZZ^\top\}$ are linearly dependent

We assume that $ZZ^\top$ can be written as a linear combination of I_2 and $YY^\top$, say,

$$ZZ^\top = \alpha I_2 + \beta YY^\top,$$

for some $\alpha, \beta \in \mathbb{R}$. Our problem becomes

$$\begin{array}{ll} \text{maximize} & \phi(a,b,c) \\ \text{subject to} & (a,b,c) \in \mathcal{S} \\ & a \leq 1 \end{array} \tag{B.12}$$

where

$$\mathcal{S} := \left\{ (a,b,c) : \exists_u : a = u^\top u, b = u^\top Y Y^\top, c = u^\top x x^\top u \right\},$$

and

$$\begin{aligned} \phi(a,b,c) \quad := \quad & a + 2\sqrt{c} + 2\sqrt{\|Y\|^2 - b + 2|\det(Y)|\sqrt{1-a}} + \\ & + 2\sqrt{\|Z\|^2 - \alpha a - \beta b + 2|\det(Z)|\sqrt{1-a}} \end{aligned}$$

is a concave function.

We have the inclusion $\mathcal{S} \subset \mathcal{T}$ where

$$\mathcal{T} := \{ (a,b,c) : \exists_{U \succeq 0} : a = \mathsf{tr}(U), b = \mathsf{tr}\left(Y Y^\top U\right), c = \mathsf{tr}\left(x x^\top U\right) \}.$$

Using $\mathcal{T}$ instead of $\mathcal{S}$ in (B.12) gives the convex problem

$$\begin{array}{ll} \text{maximize} & \phi(a,b,c) \\ \text{subject to} & a = \mathsf{tr}(U) \\ & b = \mathsf{tr}\left(Y Y^\top U\right) \\ & c = \mathsf{tr}(x x^\top U) \\ & U \succeq 0 \\ & a \leq 1 \end{array} \quad . \tag{B.13}$$

It can be shown that (B.13) can be rewritten as a SOCP. Let $U^\star$ be a solution of (B.13). Let

$$U^\star = \begin{bmatrix} u_1 & u_2 \end{bmatrix} \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \begin{bmatrix} u_1^\top \\ u_2^\top \end{bmatrix}$$

be an eigenvalue decomposition, where $\lambda_1 \geq \lambda_2$. A suboptimal solution for (B.1) is $u^\star = \pm\sqrt{\lambda_1}u_1$, where the sign is chosen such that $x^\top u^\star \geq 0$.