



INSTITUTO
SUPERIOR
TÉCNICO

UNIVERSIDADE TÉCNICA DE LISBOA
INSTITUTO SUPERIOR TÉCNICO

The Role of Rigidity in Structure from Motion: Unique Solutions in 3D Reconstruction and Object Matching

Manuel Ricardo de Almeida Rodrigues Marques

Supervisor: Doctor João Paulo Salgado Arriscado Costeira

Thesis approved in public session to obtain the PhD Degree in
Electrical and Computer Engineering

Jury final classification: Pass with Distinction

Jury

Chairperson: Chairman of the IST Scientific Board

Members of the Committee:

Doctor Hélder de Jesus Araújo
Doctor Paulo Jorge dos Santos Gonçalves Ferreira
Doctor Mário Alexandre Teles de Figueiredo
Doctor René Vidal
Doctor João Paulo Salgado Arriscado Costeira
Doctor João Manuel de Freitas Xavier

2011



INSTITUTO
SUPERIOR
TÉCNICO

UNIVERSIDADE TÉCNICA DE LISBOA
INSTITUTO SUPERIOR TÉCNICO

The Role of Rigidity in Structure from Motion: Unique Solutions in 3D Reconstruction and Object Matching

Manuel Ricardo de Almeida Rodrigues Marques

Supervisor: Doctor João Paulo Salgado Arriscado Costeira

Thesis approved in public session to obtain the PhD Degree in
Electrical and Computer Engineering

Jury final classification: Pass with Distinction

Jury

Chairperson: Chairman of the IST Scientific Board

Members of the Committee:

- Doctor Hélder de Jesus Araújo, Professor Catedrático
Faculdade de Ciências e Tecnologia da Universidade de Coimbra
- Doctor Paulo Jorge dos Santos Gonçalves Ferreira, Professor Catedrático
Universidade de Aveiro
- Doctor Mário Alexandre Teles de Figueiredo, Professor Catedrático
Instituto Superior Técnico, Universidade Técnica de Lisboa
- Doctor René Vidal, Associate Professor
Johns Hopkins University, USA
- Doctor João Paulo Salgado Arriscado Costeira, Professor Associado
Instituto Superior Técnico, Universidade Técnica de Lisboa
- Doctor João Manuel de Freitas Xavier, Professor Auxiliar
Instituto Superior Técnico, Universidade Técnica de Lisboa

Funding Institutions:
Fundação para Ciência e Tecnologia

2011

Agradecimentos

Tal como na ciência é necessário estar “sentado sobre ombros de gigantes” para romper fronteiras, eu tive e tenho o privilégio de conhecer pessoas que me possibilitaram escolher caminhos que gostaria de trilhar mas, pelo facto de ser deficiente motor, pareciam, por vezes, vedados. Continuando a analogia com o processo científico, o momento “eureka” para, após concluir a licenciatura, continuar a minha formação académica, foi o convite que me foi dirigido pelo Prof. João Sentieiro para realizar o Mestrado no Instituto de Sistemas e Robótica (ISR).

Esse convite permitiu-me realizar investigação no seio de um grupo de trabalho onde o rigor e a excelência são características genéticas. Ao longo destes últimos anos, senti sempre um grande apoio de todos os membros do ISR, nomeadamente dos Professores Maria Isabel Ribeiro e Victor Barroso que, enquanto directores, colocaram todos os meios à disposição para que pudesse desenvolver o meu trabalho com sucesso. Nesse mesmo sentido, o Instituto Superior Técnico (IST) e a Fundação para a Ciência e Tecnologia (FCT) desempenharam, enquanto instituições, um papel relevante.

Quero também destacar os meus amigos e “companheiros de caminhada” ainda de licenciatura, Duarte Antunes, Marco Morgado e Pedro Batista, e os que conheci já no ISR, Ricardo Ferreira, David Cabecinhas, Nelson Gonçalves e Paulo Rosa. Sem a sua ajuda diária e persistente, dificilmente conseguiria percorrer este caminho que agora tem o seu *terminus*.

Agradeço ao Doutor Marko Stošić os valiosos contributos que tornaram esta tese um documento mais sólido do ponto de vista teórico. Realço, igualmente, as discussões, sempre muito proveitosas, que tive com o Professor João Xavier.

Estes anos ficar-me-ão gravados na memória devido, em grande parte, ao meu orientador científico: o Professor João Paulo Costeira. Em primeiro lugar, a redacção deste documento só é possível porque o Professor João Paulo aceitou entrar nesta aventura. Tive a oportunidade de conhecer um cidadão, segundo Fernando Pessoa, “inteiro”, uma pessoa que “é todo em cada coisa”. É impossível ficar indiferente à sua energia, muitas vezes, mobilizadora de pessoas e impulsionadora de ideias. Muito para além de um orientador, sinto-me um privilegiado por ter o Professor (sim, para mim, será sempre Professor) João Paulo como amigo.

Por último, quero agradecer à minha família, especialmente aos meus pais e irmãos, pelo apoio

que, desde sempre, me têm dado. Através dos seus exemplos de vida, tenho aprendido que os obstáculos de percurso não são motivo para mudar de rumo, mas um desafio e uma oportunidade para afinar a rota traçada.

Estes são os “meus gigantes” que me permitiram alcançar esta meta. A todos, o meu muito obrigado!

Acknowledgements

As in science is necessary to be “seated on the shoulders of Giants” to break down boundaries, I also had and have the privilege to meet people who allowed me to chose paths I would like to tread but, due to the fact that I am physically disabled, seemed, at times, sealed to me. Proceeding with the analogy of the scientific process, the “eureka” moment to continue with my academic training, after graduating, was the invitation I received from Professor João Sentieiro to attain a Master Degree at Instituto de Sistemas e Robótica (ISR).

That invitation allowed me to undertake research within a working group where accuracy and excellence are genetic traits. Throughout these last years, I have always felt enormous support from all ISR members, namely Professors Maria Isabel Ribeiro and Victor Barroso whom, as ISR directors, have put all the means available so I could develop my work successfully. In the same direction, Instituto Superior Técnico (IST) and Fundação para a Ciência e Tecnologia (FCT) played a relevant role.

I also want to highlight my friends and “journey comrades”, those who come from the graduation years Duarte Antunes, Marco Morgado and Pedro Batista, and the ones I met at ISR Ricardo Ferreira, David Cabecinhas, Nelson Gonçalves and Paulo Rosa. Without their daily and persistent help, I would have hardly be able to walk this path that now meets its end.

My appreciation to Doctor Marko Stošić and his valuable contributions that made this thesis a more solid document theoretically wise. I stress, likewise, the always-helpful discussions I had with Professor João Xavier.

These years will stay engraved in my memory due to, mostly, my supervisor: Professor João Paulo Costeira. Firstly, the writing of this document was only possible because Professor João Paulo accepted to embark in this adventure. I had the opportunity to know, and quoting Fernando Pessoa, a “whole” citizen, someone “who is whole in everything”. It is impossible to remain insensitive to his energy; most of the times people-mobilizing and ideas-boosting. Far beyond a supervisor, I feel privileged to have Professor (yes, to me, it will always be Professor) João Paulo as a friend.

Lastly, I want to thank my family, specially my parents and brothers, for the support that have always given me. Through their life examples I have learned that the obstacles along the road are

not a reason to change the course but rather a challenge and an opportunity to refine it.

These are “my own giants” who allowed me to achieve this goal. To all, many thanks!

Resumo

A Reconstrução Tridimensional de uma cena a partir de imagens é um dos desafios identitários da Visão de Computador. Neste âmbito, esta tese aborda, separadamente, dois problemas: 1) reconstrução tridimensional a partir de imagens com dados omissos e 2) correspondência entre duas nuvens de pontos. As soluções propostas para ambos os problemas assentam sobre uma premissa genérica sobre a cena observada - a sua rigidez - e o modelo de câmara assumido é o afim/ortográfico.

Os algoritmos existentes para resolver o primeiro problema baseiam-se na característica da matriz dos dados. Os dados omissos são estimados impondo a restrição do número de vectores linearmente independentes. Este trabalho mostra que essa restrição não é suficiente quando, por exemplo, há imagens onde é visível apenas uma superfície planar do objecto. De modo a lidar com este tipo de dados degenerados, propõe-se uma nova factorização onde a variedade intrínseca do movimento é imposta aos dados.

Relativamente à correspondência entre conjuntos de pontos, provou-se a unicidade de solução, num espaço convexo, para o emparelhamento entre duas nuvens de pontos de igual tamanho - *Subspace Matching theorem*. Aplicando este teorema, é possível formalizar este problema como uma minimização convexa e lidar com um grande número de dados, recorrendo às ferramentas de optimização existentes. Apesar do teorema garantir solução única quando os conjuntos têm o mesmo número de pontos, os resultados mostram que esta metodologia pode ser usada em casos mais genéricos.

Palavras-Chave: Reconstrução Tridimensional, Factorização, Emparelhamento de Pontos, Optimização Convexa

Abstract

The 3D Reconstruction of a scene from images is one of the challenges that defines the Computer Vision field. In this context, this thesis addresses two issues, separately: 1) 3D reconstruction from images with missing data and 2) correspondence between two point clouds. The proposed solutions for both problems are based on a generic assumption about the observed scene - its rigidity - and we assume the affine/orthographic camera model.

The solutions for the first problem offered by the state-of-the-art algorithms are based on the rank of the matrix data. The missing data is estimated by imposing the rank constraint. This work shows that the rank condition is not enough when, for example, there are images which one single planar surface of the object is visible. In order to deal with this type of degenerate data, we propose a new factorization where the intrinsic *motion manifold* is imposed on the data.

Regarding the correspondence between sets of points, we proved the uniqueness of the solution, in a convex space, for matching between two point clouds of the same size - *Subspace Matching Theorem*. Applying this theorem, we can formalize this problem as a convex minimization and deal with large-scale data, using the optimization tools available. Although the theorem ensures a unique solution when the sets have the same number of points, the experimental results show that this methodology can be used in more general cases.

Keywords: Structure from Motion, Factorization, Point Matching, Convex Optimization

Contents

1	Introduction	1
1.1	Overview	1
1.2	A Framework for <i>Structure-from-Motion</i>	3
1.3	Problem Formulation	5
1.4	Contributions	7
1.5	Structure of the Thesis	9
2	3D Reconstruction with Missing Data	11
2.1	State-of-the-Art	12
2.2	Degenerate Data	15
2.2.1	Proposed Method	19
2.3	Experimental Results	22
2.3.1	Hotel Experiment	22
2.3.2	Dinosaur Experiment	25
2.3.3	Full reconstruction with largely scaled images	26
2.3.4	An Urban Modeling Example	28
2.4	Conclusions	30
3	The Subspace Matching Theorem: Uniqueness Conditions for Correspondence between Point Clouds	33
3.1	A Framework for Correspondence and Pose Estimation	35
3.2	Designing a Unique Solution for the Correspondence Problem	37
3.2.1	A Computational Framework	40
3.2.2	A Geometric Interpretation	40
3.2.3	Subspace Matching Theorem: a Summary of the Proof	41
3.3	Summary	45

4	Designing Optimal Search Algorithms for Large Scale Point Correspondence	47
4.1	State-of-the-Art	47
4.2	A Large-Scale Formulation	50
4.2.1	Clutter Points	52
4.2.2	Adding <i>a Priori</i> Knowledge	53
4.3	Guided Search Consensus	54
4.4	Experimental Results	56
4.4.1	Synthetic data	58
4.4.2	Real data	61
5	Conclusions	69
5.1	Discussion and Future Work	70
A	The <i>motion manifold</i> projection	73
A.1	Rigid factorization	73
A.2	The projection into scaled-Stiefel matrices	75
B	A 3D reconstruction from two images	77
C	The orthographic solutions	83
D	The subspace matching theorem	87
E	Subspace Matching: the optimization algorithm	103
E.1	The Nesterov's projected gradient algorithm	105
	Bibliography	107

List of Figures

1.1	The 3D Vision problem	1
1.2	The matching problem: the online and offline processes	2
1.3	An example of the input data	4
1.4	The Matching/Pose problem	8
2.1	Sequence of images of a rotating cube	15
2.2	The reconstruction with a rank-based algorithm vs. the proposed algorithm	16
2.3	Features in a frame sequence	17
2.4	Hotel sequence: features and degenerate data	22
2.5	Hotel sequence: pattern of missing points	23
2.6	Reconstruction results of several methods	24
2.7	The relative error of several methods	25
2.8	Dinosaur’s sequence	26
2.9	Dinosaur’s sequence results	27
2.10	The dinosaur’s 3D shape	27
2.11	An image sequence of Casa da Musica in Oporto - a “piecewise planar” building	28
2.12	Reconstruction of Casa da Musica in Oporto	29
2.13	Pattern of missing points of the urban modeling	29
2.14	A typical example of urban modeling	30
2.15	Reconstruction of an urban scenario	31
3.1	Subspace Matching problem	34
3.2	A geometric view of Subspace Matching theorem	41
4.1	<i>Shape Context</i> , an example of 2D-2D shape matching approach	48
4.2	Correspondences with elimination	52
4.3	The non-geometric costs	54
4.4	Evaluation of the GUI SAC convergence speed against two different approaches	57

4.5	Comparing two matching approaches	59
4.6	Comparing the proposed global method against a local approach	60
4.7	Performance of the matching with real data	62
4.8	A common wrong match	63
4.9	Image-to-Image matching	64
4.10	The convergence of the algorithm	65
4.11	2D-3D matching with outliers	66
4.12	2D-3D matching with clutter points	67
A.1	Projection on the <i>motion manifold</i>	73
B.1	A 3D reconstruction from two images: the used images	79
B.2	A 3D reconstruction from two images: the estimated shape	80
B.3	A 3D reconstruction from two images: Comparing with reconstruction performed from the whole image sequence	81
C.1	Two solutions	86

Chapter 1

Introduction

1.1 Overview

One of the greatest ambitions of the Computer Vision community is to devise data processing mechanisms to build up 3D models of the world from a set of photos in an automatic, easy and quick way. Such a powerful tool would have a huge impact in several distinct areas like architecture, civil engineering, robotics and the entertainment industry (e.g. movies and video games).

Using photos or video as input data to this new tool, civil engineers and architects could obtain the blueprints of existing buildings quite easily. On the other hand, the use of visual information can

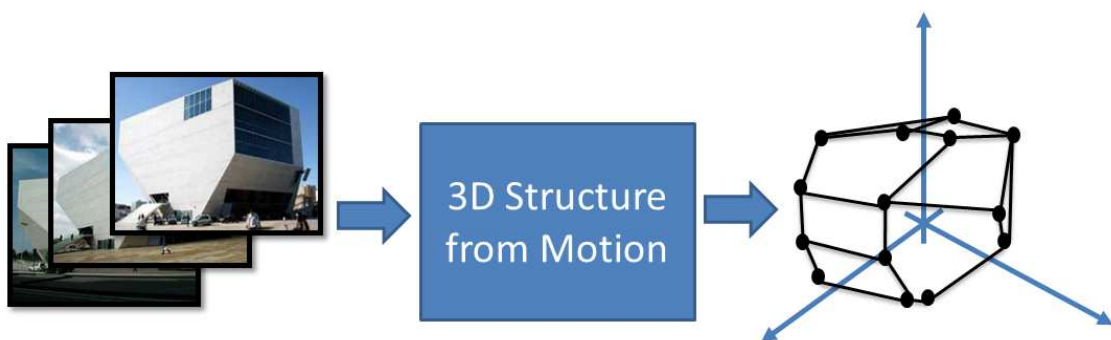


Figure 1.1: The 3D Vision problem: estimating 3D shape from a 2D image set

improve the precision of current GPS-based localization systems, as well as provide content-based capabilities. In the film industry, the creation of a movie background on images of the real world could be a much easier and faster process than it is today. The gaming and computer industry could provide more accurate real world-based scenarios for their favorite games. And finally, easy and fast computation of real scenarios can offer a great improvement in the quality of life of the general population, especially for the elderly and disabled people.

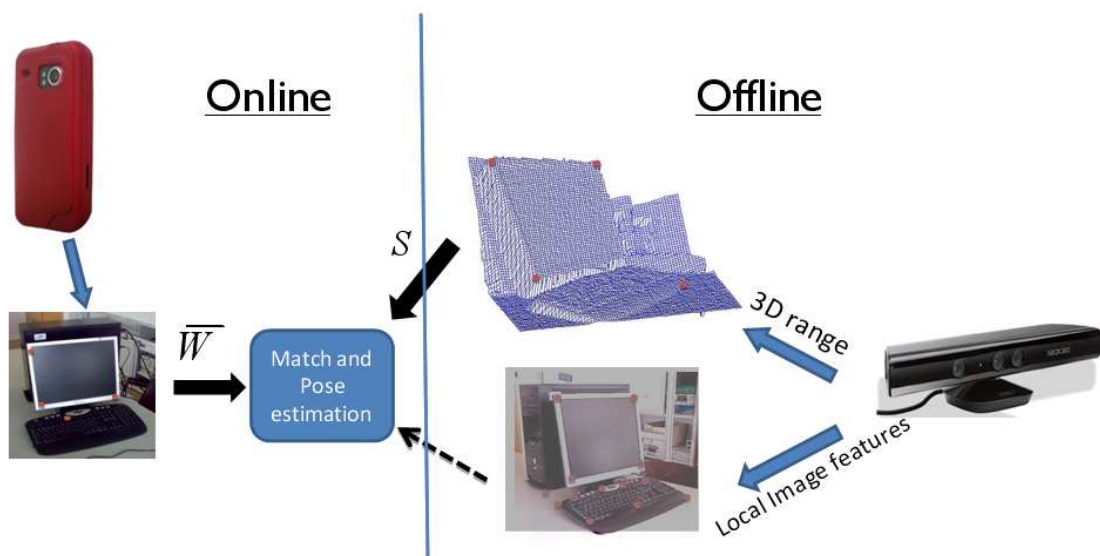


Figure 1.2: The matching problem: knowing a 3D model built offline/batch, we aim to compute a quick online match to an image captured by the device

Today, some systems exist that are able to do 3D reconstructions of large-scale scenarios [Agarwal 09], but they require massive computation to process the huge amount of data. Instead of using a data center and one day of processing time, our algorithms aim to do a 3D reconstruction (and 3D-2D identification) of a scene using low-power devices, like cell phones or tablets in fractions of a second. Figure 1.2 illustrates the concept: knowing a 3D model built offline/batch we wish to obtain a quick online match to an image captured by the device.

Knowing 2D images only, two problems have a crucial role in whole process: the 3D shape estimation of the viewed object/scene and the 3D shape recognition from a single 2D image, such

as Figure 1.1 shows. This thesis proposes computationally efficient methods to solve these two important issues. One desirable feature is the ability to deal with large amounts of data. In order to comply with this request without significant loss of generality, intrinsic geometric properties of the world, such as rigidity, are coded hinging on simplified observation models such as affine or scaled-orthographic cameras.

1.2 A Framework for *Structure-from-Motion*

The data representation and the underlying mathematical model are very important issues in designing a methodology. As Figure 1.3 suggests, the algorithms proposed in this thesis do not use all image pixels. Instead, we will rely on some relevant features (e.g. corners) that can be reliably detected. Considering a sequence, the *measurement matrix* of frame f \mathbf{W}^f is represented by

$$\mathbf{W}^f = \begin{bmatrix} u_1^f & u_2^f & \cdots & u_N^f \\ v_1^f & v_2^f & \cdots & v_N^f \end{bmatrix} \in \mathbb{R}^2, \quad (1.1)$$

where u_n^f and v_n^f are the image projection of point n .

The 3D reconstruction of a scene can be performed from one single 2D image (Figure 1.1) or from a set of images, like a video sequence. Reconstructing the 3D world from one single image, can only be done if some specific prior knowledge is available. Some examples of prior knowledge could be geometric relations between parts, known 3D coordinates of a few anchor points or global properties (e.g. planar object). On another perspective, if more images of the scene are available, the full 3D model of the object can be estimated without any prior knowledge.

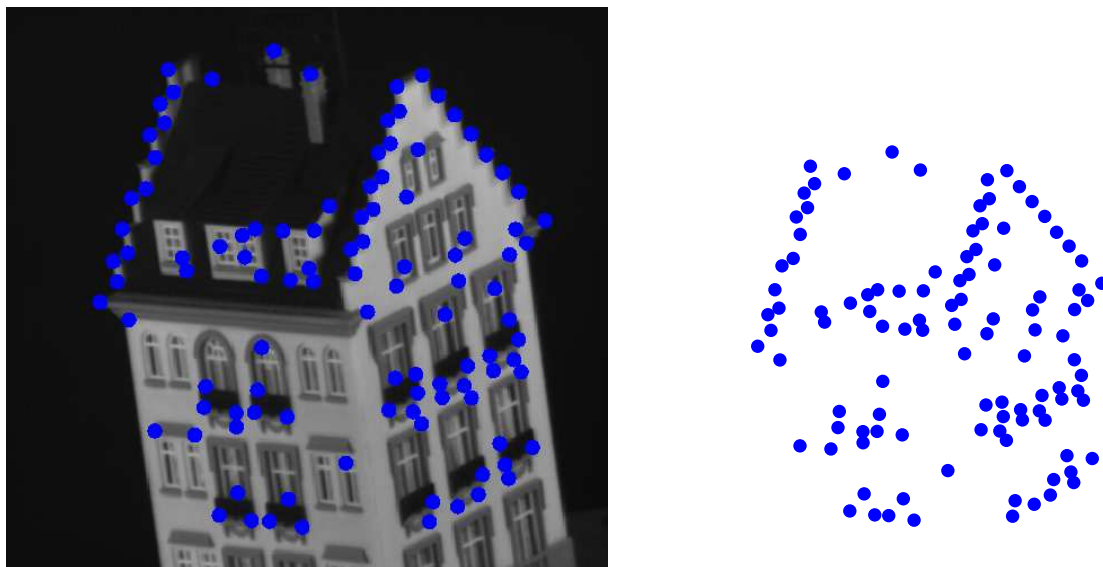


Figure 1.3: The input data: Given a 2D image only the features points locations (e.g. corners) are considered as input data

Assuming an affine/orthographic camera, the data is modeled by the following bilinear relation

$$\underbrace{\begin{bmatrix} \mathbf{W}^1 \\ \mathbf{W}^2 \\ \vdots \\ \mathbf{W}^F \end{bmatrix}}_{\mathbf{W}} = \underbrace{\begin{bmatrix} \mathbf{M}^1 & \mathbf{t}^1 \\ \mathbf{M}^2 & \mathbf{t}^2 \\ \vdots & \vdots \\ \mathbf{M}^F & \mathbf{t}^F \end{bmatrix}}_{\begin{bmatrix} \mathbf{M} & \mathbf{t} \end{bmatrix}} \begin{bmatrix} \mathbf{S} \\ \mathbf{1}^T \end{bmatrix} \in \mathbb{R}^{2F \times N}, \quad (1.2)$$

where \mathbf{W} is the *measurement matrix*, \mathbf{M} the *motion matrix* and \mathbf{t} the *translation vector*. The *shape matrix* \mathbf{S} is composed of N 3D points as the following equation shows:

$$\mathbf{S} = \begin{bmatrix} x_1 & x_2 & \cdots & x_N \\ y_1 & y_2 & \cdots & y_N \\ z_1 & z_2 & \cdots & z_N \end{bmatrix} \in \mathbb{R}^{3 \times N} \quad (1.3)$$

The world 3D shape \mathbf{S} can be computed from a set of 2D images \mathbf{W} using a well-known factorization method proposed by [Tomasi 92]. This approach recovers the *shape matrix* in a two step fashion. First, the mean subtracted measurement matrix is factorized subject to rank 3. Then, an affine map of shape is computed by imposing the orthogonality properties of *motion matrix* \mathbf{M} (see Appendix B).

Many computer vision problems have reliable solutions rooted in this model (1.2). That is the case of the factorization method [Tomasi 92], *multi-body segmentation* [Costeira 97], *structure-from-motion* with uncertainty [Anandan 02] and *non-rigid motion* [Del Bue 06]. A recent work extended this framework to manage with deformable surfaces [Ferreira 09]. The same bilinear data model is used for other purposes: eigenfaces [Turk 91], shape from shading [Prados 05], style and content [Tenenbaum 00], structure from sound [Thrun 05] and photometric stereo [Basri 07], to name a few.

Although this bilinear model is a fundamental tool in the computer vision field, it is not the most accurate model. The affine or orthographic cameras modeled by (1.2), though an approximate solution of perspective cameras, enable simple and computationally efficient algorithms. Given the simplification assumed by these camera models, the approaches proposed in this thesis are valid when the object size is small compared to its distance to the camera (see details in [Hartley 03b]).

1.3 Problem Formulation

As we referred to earlier, our 3D reconstruction approach requires a set of several 2D images, where the considered object can be viewed from different viewpoints. In many real situations, the factorization method [Tomasi 92] does not provide the solution because the input images have missing data due to self-occlusion, occlusion by other objects and errors of feature detection. Another issue to consider, which naturally arises when more than one image is used, is a correspondence problem between points of different images.

A Global View:

Although we will consider *3D reconstruction with missing data* and *point matching* as different problems, the following optimization problem shows that both problems share the same structure. Shape, motion and point correspondences are the minimizers of the following optimization problem,

$$\begin{aligned}
 (\mathbf{S}, \mathbf{M}^f, \mathbf{P}_{p_f}^f, \mathbf{t}_f)^* = & \arg \min_{\mathbf{S}, \mathbf{M}^f, \mathbf{P}_{p_f}^f, \mathbf{t}_f, \mathbf{D}^f, p_f} \sum_{f=1}^F \left\| \overline{\mathbf{W}}^f \mathbf{P}_{p_f}^f - \begin{pmatrix} \left[\mathbf{M}^f \quad \mathbf{t}^f \right] \begin{bmatrix} \mathbf{S} \\ \mathbf{1}^T \end{bmatrix} \end{pmatrix} \odot \mathbf{D}^f \right\|_2^2 \\
 \text{s.t.} & \mathbf{P}_{p_1}^1, \mathbf{P}_{p_2}^2, \dots, \mathbf{P}_{p_F}^F \in \mathcal{P}_{p_f} \\
 & \mathbf{M}_1, \mathbf{M}_2, \dots, \mathbf{M}_f \in \mathcal{S}
 \end{aligned} \tag{1.4}$$

where the unknown variables $\mathbf{P}_{p_f}^f$ belong to the set of $N \times p_f$ partial permutation matrices \mathcal{P}_{p_f} , \mathcal{S} is a generic set¹ and the symbol \odot is the Hadamard (elementwise) product. The value p_f is the number of possible (correct) correspondences between image f and 3D shape \mathbf{S} . The matrix $\overline{\mathbf{W}}^f$ is a permuted version of \mathbf{W}^f . The matrix \mathbf{D}^f , which is called *mask matrix* of frame f , indicates if a 3D point on the considered shape is visible in image f , i. e.

$$\mathbf{D}_{[1,i]}^f = \mathbf{D}_{[2,i]}^f = \begin{cases} 1, & \text{if 3D point } i \text{ of } \mathbf{S} \text{ is visible} \\ 0, & \text{otherwise} \end{cases} .$$

Problem (1.4) is clearly an ill-posed problem but puts in evidence the similar structure of the two main issues we wish to address. In conclusion, we will tackle the 3D reconstruction with missing data and point cloud correspondence as two different subproblems. These subproblems result from breaking down Problem (1.4) and including some additional constraints.

¹Depending on the context, \mathcal{S} can be the set of affine matrices or the stiefel manifold.

1.4 Contributions

In this thesis, we propose new methodologies to address the following questions:

1. How to estimate the 3D shape of a generic scene from image sequences with partial views?
2. How to match 3D and 2D objects from one single image? How can object pose be computed from this single image?

As we said before, our answers to both questions are framed as optimization problems derived from expression (1.4), assuming some variables are known. If \mathbf{D}^f and $\mathbf{P}_{p_f}^f$ matrices are known, the 3D shape can be computed from sequences of images with missing data by solving the following problem

$$\begin{aligned}
 (\mathbf{S}, \mathbf{M}, \mathbf{t})^* = \arg \min_{\mathbf{S}, \mathbf{M}, \mathbf{t}} & \left\| \left(\mathbf{W}' - \begin{bmatrix} \mathbf{M} & \mathbf{t} \end{bmatrix} \begin{bmatrix} \mathbf{S} \\ \mathbf{1}^T \end{bmatrix} \right) \odot \mathbf{D} \right\|_2^2 \\
 \text{s.t. } & \mathbf{M} \in \mathcal{M}
 \end{aligned} \tag{1.5}$$

where \mathbf{W}' is a *data matrix* where its entries are equal to \mathbf{W} if the correspondent points are visible. The set \mathcal{M} is the *motion manifold* and a matrix belongs to it if is composed by a stack of stiefel matrices. In order to solve the first question, our main contributions are:

- An iterative algorithm to compute the 3D shape from missing data with generic pattern (Chapter 2).
- A new factorization method that imposes the orthogonality constraints to the *motion matrix* (Section 2.2.1 Appendix A).

Both solutions are rooted in a new *motion projection* (Appendix A.2). The new factorization method enables the reconstruction of a 3D object from one single pair of images, assuming the orthographic model (Appendix B).

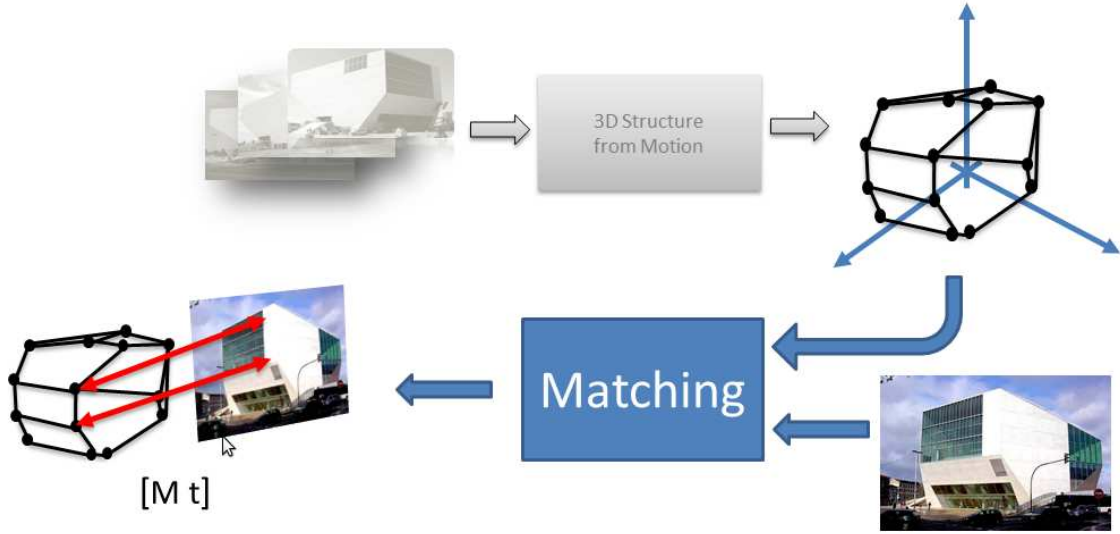


Figure 1.4: The Matching/Pose problem: given a model and an image, the goal is to find the pose $[\mathbf{M} \ \mathbf{t}]$ and the correspondence between model and image points

After estimating the 3D shape of an object or a scene, the shape recognition from a single 2D and the camera orientation estimation, as Figure 1.4 suggests, can be computed solving the following problem:

$$\begin{aligned}
 (\mathbf{M}, \mathbf{P}^f, \mathbf{t}^f)^* &= \arg \min_{\mathbf{M}, \mathbf{P}^f, \mathbf{t}^f} \sum_{f=1}^F \left\| \overline{\mathbf{W}}^f \mathbf{P}^f - \begin{bmatrix} \mathbf{M}^f & \mathbf{t}^f \end{bmatrix} \begin{bmatrix} \mathbf{S} \\ \mathbf{1}^T \end{bmatrix} \right\|_2^2 \\
 \text{s.t.} & \quad \mathbf{P}^1, \mathbf{P}^2, \dots, \mathbf{P}^F \in \mathcal{P} \\
 & \quad \mathbf{M} \in \mathcal{S}
 \end{aligned} \tag{1.6}$$

This optimization problem has a similar structure to (1.4) but, instead of the set of the partial permutation set, the solution belongs to the set of permutations matrices \mathcal{P} . In this case, the shape \mathbf{S} is known and all its points and projections are visible (*the mask matrix* \mathbf{D} disappears because we do not consider missing data in the matching problem).

The main challenges of a generic correspondence problem are its combinatorial nature and the large number of variables. In order to face these issues, our contributions are:

- A theorem² that guarantees a unique solution for the correspondence problem if the data is modeled by a linear subspace (Chapter 3 and Appendix D). Using this theorem, we are able to find the correspondence between two set of points through convex optimization (Section 3.2.1).
- A computationally efficient implementation for the correspondence problem based on the gradient projection framework (Chapter 4 and Appendix E).
- A guided search mechanism to speed up the convergence rate, increase the resilience to outliers and deal with clutter points. This approach is based on the affine transformation between the two point clouds, which can be defined by a small set of correspondences (Section 4.3).

1.5 Structure of the Thesis

The document is organized as follows:

- In Chapter 2, the 3D reconstruction with missing data problem is discussed. A large number of state-of-the-art methods use the rank as the constraint of the *motion matrix*. When we have degenerate data, this model is not able to reconstruct a 3D shape. To cope with degenerate data, new constraints are used and a new factorization method is presented. The performance of our methodology is evaluated with synthetic and real experiments.
- In Chapters 3 and 4, we tackle the correspondence problem. Based on a theoretical principle, we propose, in Chapter 3, a new formulation that allows us to compute the optimum matching between two point clouds belonging to the same subspace. The solution can be found by solving a convex optimization problem.

²The proof (Appendix D) was derived in collaboration with Marko Stošić and published in [Stosic 11].

- Due to the huge number of variables, a minimization process of the convex program is proposed in Chapter 4. The followed strategy builds upon a state-of-the-art minimization algorithm and an EM-scheme based on the affine transformation between the two point clouds. According to the experiments presented in this chapter, the developed method increases the convergence rate significantly.
- Chapter 5 draws the conclusions and discusses some future research directions to cope with outliers. Possible ways to improve the performance of geometric-based methods using photometric information are also discussed.

Chapter 2

3D Reconstruction with Missing Data

As we referred to earlier, the mathematical formulation of 3D Reconstruction with missing data problem can be given by the following optimization problem:

$$\text{Problem 1} \quad (\mathbf{S}, \mathbf{M}, \mathbf{t})^* = \underset{\mathbf{S}, \mathbf{M}, \mathbf{t}}{\operatorname{arg\,min}} \left\| \left(\mathbf{W}' - \begin{bmatrix} \mathbf{M} & \mathbf{t} \end{bmatrix} \begin{bmatrix} \mathbf{S} \\ \mathbf{1}^T \end{bmatrix} \right) \odot \mathbf{D} \right\|_2^2$$

s.t. $\mathbf{M} \in \mathcal{S}$

, where the *mask matrix* \mathbf{D} , the *translation vector* \mathbf{t} , which result by concatenating similar entities from 1.4, and \mathbf{W}' are given by:

$$\mathbf{D} = \begin{bmatrix} \mathbf{D}^1 \\ \mathbf{D}^2 \\ \vdots \\ \mathbf{D}^F \end{bmatrix}, \quad \mathbf{t} = \begin{bmatrix} \mathbf{t}^1 \\ \mathbf{t}^2 \\ \vdots \\ \mathbf{t}^F \end{bmatrix},$$

$$\mathbf{W}' = \mathbf{W} \odot \mathbf{D}.$$

The projection set \mathcal{M} , the *motion manifold*, is defined as $\{\mathbf{M} \in \mathcal{M} : \forall_f \mathbf{M}^f (\mathbf{M}^f)^T = \alpha_f \mathbf{I}, \alpha_f \in \mathbb{R}\}$. Note that \mathbf{W}' has N columns and no visible features are replaced by zeros.

As we will show, several methods have been proposed to solve Problem 1, which recover the *shape* and *motion* matrices. But when we are in the presence of degenerate data, these state-of-the-art approaches do not guarantee the correct solution because the assumed data model does not cope with degeneracies. In this chapter, we discuss this *phenomenon* and propose an approach to cope with this type of data.

2.1 State-of-the-Art

The Tomasi-Kanade factorization method [Tomasi 92] fired an enormous body of work in the area of structure-from-motion. After the initial developments focused mainly on the 3D reconstruction algorithms, two key challenges received great attention in the past years:

- Reconstruction with missing data : Designing structure-from-motion algorithms that handle partial views of the objects
- 3D reconstruction from sequences with degenerate motion and/or degenerate shapes.

To compute the 3D shape from missing data, some approaches have been proposed in the last two decades. According to the orthographic camera model, matrix \mathbf{W} has rank 4 and this fact is used to estimate the unknown data. In the original factorization method [Tomasi 92], authors suggested that the missing data could be sequentially replaced using complete subsets of the data. But, the problem of finding the largest sub-matrix with missing elements is NP-hard [Jacobs 97]. This first approach does not solve a generic missing data problem, as proved by Jacobs. This means that the imputation algorithm proposed in [Tomasi 92], solves the problem only for a subset of possible missing data pattern configurations (matrix \mathbf{D} Problem 1).

Matrix completion with rank constraints

The factorization method hinges on the rank 4, constraint of the data. This constraint is widely used in the state-of-the-art and constitutes the most relevant criterion behind recent developments.

In one of the earliest works [Jacobs 97], Jacobs proposes a non-iterative and sub-optimal algorithm where the measurement matrix verifies the rank constraint. The completion results from the

intersection of affine subspaces generated by the missing entries in collections of 4 columns (rank 4). In the same way as Jacobs', there are several algorithms known as *batch algorithms* [Guilbert 06, Martinec 05, Tardif 07, Triggs 97], because the solution (sub-optimal in presence of noise) is found in one global step. In [Rao 10], the missing data entries of a matrix are estimated through sparse reconstruction of subspaces. Subspace projections are computed using the ℓ_1 norm approximation to the ℓ_0 norm.

In order to refine the solution obtained by *batch algorithms*, *iterative algorithms* play an important role, namely *alternation algorithms*. These last algorithms are based in the fact that if \mathbf{M} or \mathbf{S} are known, there is a closed-form solution for the other such that Problem 1 is minimized. This can be seen as an EM scheme [Roweis 97].

Alternation algorithms seek a better solution (a lower value for the cost function of Problem 1) than the first ones. The Wiberg algorithm proposed in [Wiberg 76] solves the factorization problem with missing data through an alternation scheme. Extensions to this approach were presented in [Shum 95], where the mask matrix \mathbf{D} can have arbitrary weighting values as well as in [Hartley 03a], where authors suggest adding a normalization step between the two factors updates. A generalization of the Wiberg algorithm is presented by [Eriksson 10], allowing us to solve this missing data problem with outliers. In order to cope with outliers, this method uses a common optimization technique and reformulates a ℓ_2 norm problem into a ℓ_1 problem, such as [Ke 05]. The Guerreiro and Aguiar approach [Guerreiro 03] is similar to Aanaes [Aanaes 02], since both algorithms, at each iteration, project the data in a rank-4 subspace using the ℓ_2 norm. The convergence of the referred methods is initially good but it is very susceptible to flatlining. The Buchanan and Fitzgibbon work [Buchanan 05] proposed a Newton method to improve convergence, and it is today one of the most accurate and robust rank-constrained algorithms.

Recently, several new approaches present closed-form solutions or convex approximations to the rank constraint, allowing precise and efficient computation of global optimum estimates. If the missing data pattern form a Young diagram and if the completion criterion to be minimized is the k^{th} singular value, [Aguiar 08] shows that the optimum estimate can be computed in closed form.

Having in mind the rank constraint, [Cai 10] uses the matrix nuclear norm as an approximation to its rank, thus attaining a global optimum. Several methods combine nuclear and ℓ_1 norms in a similar fashion in order to estimate the missing data. Finally, by noting that the data matrix can be generated by a low rank model added to a sparse error matrix, Candes and Wu [Candès 11, Wu 11] developed convex programs using simultaneously the above approximations to the rank and sparsity. Thus a low rank factorization is obtained in data sets contaminated with outliers.

It is important to note that all these algorithms use *à priori* knowledge about the rank of the measurement matrix (in this case rank 4) to estimate the completion. However, the rank constraint is not adequate to obtain a correct estimate when images in the sequence have all visible points belong to 1D or 2D subspaces (lines and planes). Gross errors happen because the optimization Problem 1 has infinite *minimae*.

Handling degeneracies in SfM

Since the data model is constrained by its rank, in situations where the motion or the object shape are of lower rank, most of the above algorithms become ambiguous. Some works try to solve this ambiguity by including more *à priori* information. In [Sparr 98], the planar surfaces of the object are known and due to this they impose these constraints to the object's shape. On the other hand, if motion does not span all degrees of freedom (e.g. rotation around one axis), observations will lie on lower dimensional spaces, thus producing "degeneracies". In [?, Vidal 02] planar motion degeneracies are handles by explicitly enforcing the degenerate model. Another approach considers a smooth camera trajectory [Guilbert 06, Tardif 07]. If the missing data matrix \mathbf{D} is a block-diagonal matrix, we can find the best model (planar or non-planar) to explain each part of the object [Kanatani 98]. Recently, [Ferreira 09] proposed a general framework in order to reconstruct planar and deformable surfaces.

In summary, the state-of-the-art handles occlusion in 3D reconstruction recurring to the rank as the criterion to fill the missing entries. On the other hand, degeneracies are handled by explicitly coding them in the algorithms. If both events occur, as we will show, no method exists guaranteeing a acceptable reconstructions.

In this chapter, we will present a methodology that tackles both issues simultaneously, allowing the reconstruction of 3D objects from degenerate sequences. Very recently, [Del Bue 10] generalized this framework to any bilinear model using Augmented Lagrangian Multipliers.

2.2 Degenerate Data

A correct solution for the 3D shape estimation from a generic missing data pattern is not achieved using the mentioned strategies. The used constraints by rank-based methods are not enough to obtain a correct reconstruction and the latter methodologies need some *à priori* information about camera motion or shape attributes. We will show that the 3D shape can be computed by solving Problem 1 and constraining camera motion to comply with the rigidity constraints. To illustrate the relevance of the problem, consider the simple synthetic example depicted in Figure 2.1.

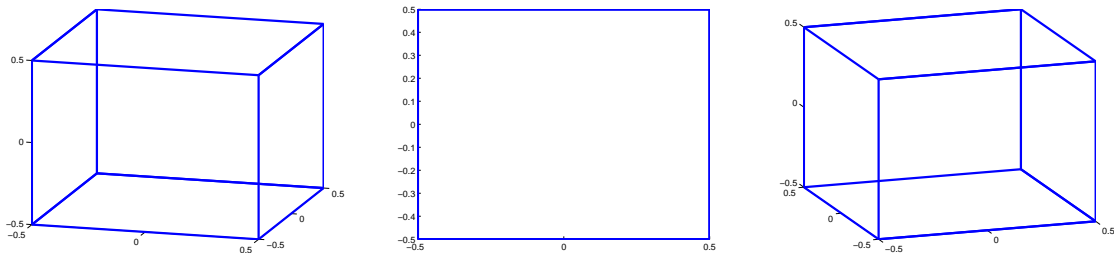


Figure 2.1: Sequence of images of a rotating cube (3 samples). Features are the visible vertices. The frontal view of the middle image is degenerate.

In this case, the considered object is a cube and feature points are its vertices. We generated an artificial sequence of 10 images of a rotating cube. One of the images is a frontal view where visible features belong to the one single face of the cube. Figure 2.2 shows the resulting shape (in red), after completing the missing vertices with one of the state-of-the-art algorithms. The green cube is the result of our algorithm totally coincident with ground truth (no noise). Note that the completed observation matrix was rank 4.

For a better reader's understanding about this issue, we will describe the general framework

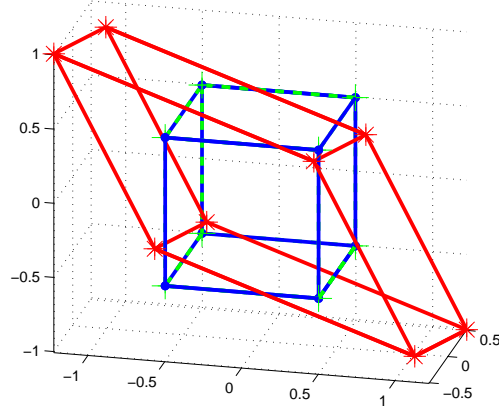


Figure 2.2: The reconstruction with a rank-based algorithm (red) vs. the proposed algorithm (blue)

of rank-based methods and the reasons why imposed constraints are not enough to estimate the (correct) 3D shape. In the end, we present an approach to fix it.

All rank-based works mentioned above propose different algorithms to find a solution for the following problem:

$$\begin{aligned}
 (\mathbf{M}, \mathbf{S})^* &= \arg \min_{\mathbf{M}, \mathbf{S}} \|\mathbf{W}' - \mathbf{MS}\|_F^2 \\
 &\text{s.t. } \mathbf{M}, \mathbf{S} \in \mathcal{R}_4
 \end{aligned} \tag{2.1}$$

where \mathcal{R}_n is the space of matrices with rank equal to n .

Using this framework, the solution obtained is unique if it exists, at least, one image where the known points are in the same 1D or 2D subspace (degenerate images). This happens because problem (2.1) has an infinite number of solutions, that is, the non-zero values of \mathbf{W}' (the visible features) can be obtained from different rank 4 matrices \mathbf{M} and \mathbf{S} .

To verify this fact, consider the *translation vector* as a zero vector and let \mathbf{W}'_c be given by

$$\mathbf{W}'_c = \mathbf{MS} \odot \mathbf{D}$$

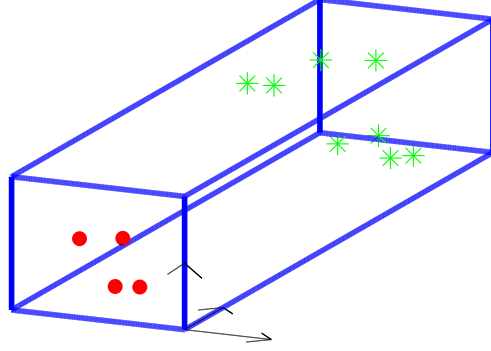


Figure 2.3: Features in frame f : the known points in frame f are represented by red circles and the unknown points by green stars

, where $\mathbf{W}'_c \in \mathcal{R}_3$. Suppose that the first m points belong to a 2D flat surface and that the known points of the image f are precisely these points (see fig. 2.3). The object's shape is known and we can calculate a matrix \mathbf{M}^f such that the plane $z = 0$ contains the first m points.

$$\underbrace{\begin{bmatrix} (\mathbf{W}'_c)_{[2f-1,1]} & \cdots & (\mathbf{W}'_c)_{[2f-1,m]} & ? & \cdots & ? \\ (\mathbf{W}'_c)_{[2f,1]} & \cdots & (\mathbf{W}'_c)_{[2f,m]} & ? & \cdots & ? \end{bmatrix}}_{\mathbf{W}'_c} = \underbrace{\begin{bmatrix} a & b & c \\ d & e & f \end{bmatrix}}_{\mathbf{M}^f} \mathbf{A} \quad (2.2)$$

$$\mathbf{A}^{-1} \underbrace{\begin{bmatrix} \mathbf{S}_{11} & \cdots & \mathbf{S}_{1m} & \mathbf{S}_{[1,m+1]} & \cdots & \mathbf{S}_{1N} \\ \mathbf{S}_{21} & \cdots & \mathbf{S}_{2m} & \mathbf{S}_{[2,m+1]} & \cdots & \mathbf{S}_{2N} \\ 0 & \cdots & 0 & \mathbf{S}_{[3,m+1]} & \cdots & \mathbf{S}_{3N} \end{bmatrix}}_{\mathbf{S}}$$

This particular parametrization just makes it completely clear that the submatrix of the first m columns of \mathbf{S} is singular (the planar surface). Equation (2.2) is equivalent to the equation system

given by

$$\left\{ \begin{array}{l} (\mathbf{W}'_c)_{[2f-1,1]} = a\mathbf{S}_{11} + b\mathbf{S}_{21} \\ (\mathbf{W}'_c)_{[2f,1]} = d\mathbf{S}_{11} + e\mathbf{S}_{21} \\ \vdots = \vdots \\ (\mathbf{W}'_c)_{[2f-1,m]} = a\mathbf{S}_{1m} + b\mathbf{S}_{2m} \\ (\mathbf{W}'_c)_{[2f,m]} = d\mathbf{S}_{1m} + e\mathbf{S}_{2m} \end{array} \right. \quad (2.3)$$

In the system of equations (2.3) we can clearly verify that Problem 1 has two degrees of freedom, variables c and f . Then, the unknown image points projection in frame f have not two solutions, but infinite. This fact does not allow us to recover the original *motion* and *shape matrices* because the method can add an error to the known points' projections in this frame. In other words, there can be a large error in the image position (due an arbitrary c and f parameters) and yet the (completed) observation matrix will be rank 4 and not conformal with the orthographic camera model. In real situations with noise and other distortions these estimated projections can be quite far from reality. As it is common in real situations, there are large sets of degenerate images and the total error can be significant. In summary, regardless of the algorithm we may use to fill the missing values, there are two free variables which will distort the track matrix and thus distort the shape and the motion.

This situation can be easily overcome by imposing the orthogonality constraints for each *camera matrix*. In our case, the *camera matrix* \mathbf{M}^f (2.2) has its entries constrained as follows:

$$\left\{ \begin{array}{l} a^2 + b^2 + c^2 = \alpha^2 \\ d^2 + e^2 + f^2 = \alpha^2 \\ ad + be + cf = 0 \end{array} \right. \quad (2.4)$$

where α is the scale factor for that frame. Adding the orthogonality constraints (2.4) to the equation

system (2.3), variables c and f have only the two following solutions :

$$c = \pm \sqrt{\frac{1}{2} \left(-a^2 - b^2 + d^2 + e^2 + \sqrt{gh} \right)} \quad (2.5)$$

$$f = \mp \sqrt{\frac{1}{2} \left(a^2 + b^2 - d^2 - e^2 + \sqrt{gh} \right)} \quad (2.6)$$

where $g = d^2 + 2db + b^2 + e^2 - 2ea + a^2$ and $h = d^2 - 2db + b^2 + e^2 + 2ea + a^2$.

As expected the rank ambiguity is resolved and the solution is rigid. For further details see Appendix A. The two algebraic solutions correspond to a reflection on the image plane which are compatible with the scaled-orthographic model.

2.2.1 Proposed Method

As stated before, the degenerate data *phenomenon* does not depend on the used optimization algorithms but have deeper roots: the optimization problem formulation and more precisely the admitted camera model. Checking equations (2.3), we easily conclude that there are infinite solutions because the affine camera model is accepted by the rank methodology. Due to this reason, we replace Problem 1 by the following one, redefining the camera constraints.

$$(\mathbf{M}, \mathbf{t})^* = \underset{\mathbf{M}, \mathbf{t}}{\operatorname{arg\,min}} \left\| \left(\mathbf{W}' - \begin{bmatrix} \mathbf{M} & \mathbf{t} \end{bmatrix} \begin{bmatrix} \mathbf{S} \\ \mathbf{1}^T \end{bmatrix} \right) \odot \mathbf{D} \right\|_F^2$$

$$\mathbf{M}^1 (\mathbf{M}^1)^T = \alpha^1 \mathbf{I}$$

Problem 2

$$\begin{aligned} & \vdots \\ \text{s.t.} & \mathbf{M}^F (\mathbf{M}^F)^T = \alpha^F \mathbf{I} \\ & \alpha^f \in \mathbb{R}^+, \forall_f \end{aligned}$$

Note that this last problem is exactly that of equation (1.5). In this case, we explicit the *motion manifold* \mathcal{M} in the optimization problem constrains. Instead of the infinite number of solutions, only two solutions (the original solution and the reflected one) comply to the orthographic constraints.

To find a solution for Problem 2 we propose the algorithm described below.

Algorithm 1 Rigid Factorization with Missing Data

1. Initializations: $\mathbf{W}'_0 = \mathbf{W}$, $k = 0$
2. Estimate translation (centroid).
 $\mathbf{t}_k = \left[\frac{1}{P} \sum_i (\mathbf{W}'_{1i})_k \quad \cdots \quad \frac{1}{N} \sum_i (\mathbf{W}'_{[2f,i]})_k \right]$
 $(\mathbf{W}'_c)_k = \mathbf{W}'_k - \mathbf{t}_k$ Remove translation
 $k = k + 1$
3. Estimate \mathbf{M}_k and \mathbf{S}_k Using Rigid Factorization
4. Update data matrix

$$\mathbf{W}'_k = \underbrace{(\mathbf{M}_k \mathbf{S}_k + \mathbf{t}_{k-1} \mathbf{1}_{[2F,N]}) \odot \bar{\mathbf{D}}}_{\text{Missing data estimate}} + \underbrace{\mathbf{W}' \odot \mathbf{D}}_{\text{Known data}}$$
 $\bar{\mathbf{D}}$ - 2's complement of \mathbf{D} i.e. $\bar{\mathbf{D}} = \mathbf{1}_{[2F,N]} - \mathbf{D}$
5. Verify if $\|\mathbf{W}'_k - \mathbf{W}'_{k-1}\| < \epsilon$.
 If not verify go to step 2 and $k = k + 1$.

Our optimization method is very similar to some rank-based approaches (e.g. [Guerreiro 03]). In step 1 (Initialization), the entries missing in data matrix \mathbf{W}' are initialized with random values in the range of the known entries. In each iteration, the *translation vector* is composed by the mean point of each frame¹.

The main difference between Algorithm 1 and rank-based methods is in step 3, concerning to the computation of \mathbf{M} . In our case, the orthographic camera model is fully imposed by solving, at each iteration, the optimization problem expressed by

$$(\mathbf{M})^* = \underset{\mathbf{M}}{\operatorname{arg\,min}} \sum_{f=1}^F \|\mathbf{W}'_c^f - \mathbf{M}^f \mathbf{S}\|_F^2$$

$$\mathbf{M}^1 (\mathbf{M}^1)^T = \alpha^1 \mathbf{I}$$

Problem 3

$$\begin{aligned} & \vdots \\ \text{s.t.} \quad & \mathbf{M}^F (\mathbf{M}^F)^T = \alpha^F \mathbf{I} \\ & \alpha^f \in \mathbb{R}^+, \forall f \end{aligned}$$

and can be solved using the following algorithm

¹If one point is visible in whole video sequence, the *translation vector* is not estimated.

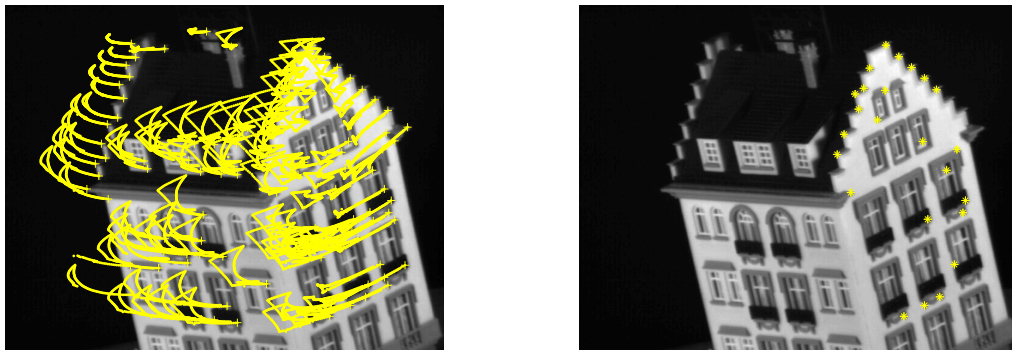
Algorithm 2 Rigid Factorization

1. Initializations:
 (factorize \mathbf{W}'_c using any factorization (e. g. SVD))
 $\mathbf{W}'_c = \mathbf{A}\mathbf{B}$, $\mathbf{R} = \mathbf{A}$, $\mathbf{M}_0 = \mathbf{A}$, $\mathbf{S}_0 = \mathbf{B}$
 $k = 1$
 2. Project \mathbf{R} into the manifold of motion matrices
 $\mathbf{M}_k = \arg \min_{\mathbf{X}} \sum_f \|\mathbf{R}^f - \mathbf{X}^f\|_F^2$
 s. t. $\mathbf{X}^f \mathbf{X}^{fT} = \alpha^f \mathbf{I} \quad \forall f$
 $\alpha \in \mathbb{R}^+$
 3. $\mathbf{S}_k = \mathbf{M}_k^+ \mathbf{W}'_c$, \mathbf{M}_k^+ - Moore-Penrose pseudoinverse
 4. $\mathbf{R} = \mathbf{W}'_c \mathbf{S}_k^+$
 5. Verify if $\|\mathbf{M}_k - \mathbf{M}_{k-1}\| < \epsilon$.
 If not, go to step 2 and $k = k + 1$.
 6. $\mathbf{M} = \mathbf{M}_k$ and $\mathbf{S} = \mathbf{S}_k$
-

Note that here there is no missing entries. Given a *data matrix* \mathbf{W}_c , the *Rigid Factorization* computes matrices \mathbf{A} and \mathbf{B} such that $\mathbf{W}_c = \mathbf{A}\mathbf{B}$ and \mathbf{A} is a piecewise Stiefel matrix. Note that the projection of each matrix \mathbb{R}^f on the Stiefel manifold (step 2) has closed-form (see details in Appendix A).

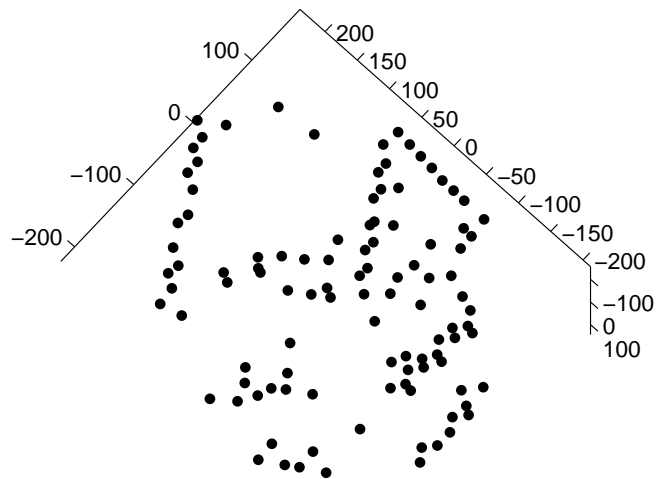
Solving Problem 3, we are able to estimate the 3D shape from two images without missing data, as opposed to the Tomasi-Kanade algorithm [Tomasi 92] which requires, at least, three images (see Appendix B).

Since the proposed algorithm for 3D shape estimation with missing data is iterative, convergence is an essential topic while evaluating its performance. We did some synthetic experiments to illustrate the algorithm's behavior in different situations. The obtained results are presented in the following section.



(a) Feature tracks

(b) Degenerate frames



(c) Hotel 3D shape

Figure 2.4: Hotel sequence: features and degenerate data

2.3 Experimental Results

2.3.1 Hotel Experiment

Benchmark tests were performed against the state-of-the-art method, using Buchanan & Fitzgibbon's Matlab code². To prevent a full match between the data and our model we only used real

²This package is available in www.robots.ox.ac.uk/~abm

data. We modified the known hotel sequence³, selecting all 106 feature points and 18 equally spaced frames from the total of 180. A random pattern of missing features was generated with only 14% missing features (Figure 2.5). Two of those frames were artificially made degenerate with only 24 features visible, all lying on a planar surface (the rightmost wall of the hotel, shown in Figure 2.4b).

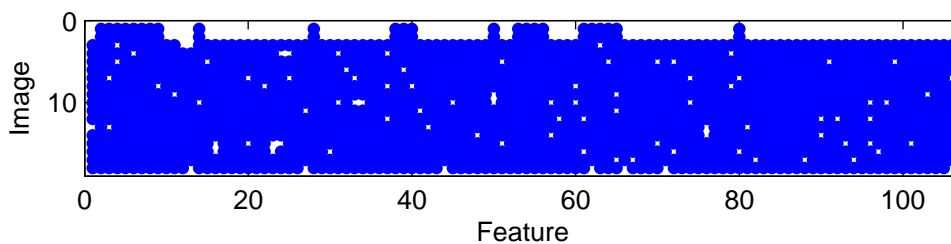


Figure 2.5: Hotel sequence: pattern of missing points. First 2 images are degenerate

Since there is no ground-truth, we used the object’s shape computed using Tomasi-Kanade’s factorization method with full observation matrix (no missing data), as the reference shape (Figure 2.4c).

In Figures 2.6 and 2.7 we show the performance of our (rigid factorization) algorithm against 4 state-of-the-art methods: BF-Buchanan&Fitzgibbon’s Damped Newton method [Buchanan 05], PF-power factorization [Hartley 03a, Vidal 04], GA-Guerreiro&Aguiar EM (alternate) algorithm [Guerreiro 03], Aanaes-Aanaes et al [Aanaes 02]. To avoid graphical clutter, in Figure 2.6 we show the reconstruction of two methods (GA and BF), the “ground-truth” (TK) and our method (RF). Note that an Euclidean upgrade was performed to all matrix completion algorithms to obtain a 3D reconstruction. The figure was generated from a top view so that the 3 planes of the hotel walls can be perceived clearly. Recall that GA and BF are quite different in nature (an alternate algorithm vs. a Newton algorithm). Since both methods seek the best rank 4 matrix they reach similar solutions, both inadequate. As expected by enforcing rigidity constraints the reconstruction is quite close to

³<http://vasc.ri.cmu.edu/idb/html/motion/index.html>

the reference. All other methods behaved in much the same way, and this is reflected in the error plot shown in the graph of Figure 2.7. Here, each bar represents the percentage of shape error $(\sum |S_i - \hat{S}_i^{tk}|) / (\max(S^{tk}) - \min(S^{tk}))$, where \hat{S}_i is the shape estimate for point i and S_i^{tk} is the reference shape for that point. Of course the absolute value of the error depends on the particular shape, but the relevant aspect is that all other methods obtain estimates that are at least one order of magnitude above our method, and this happens with only 14% of data missing.

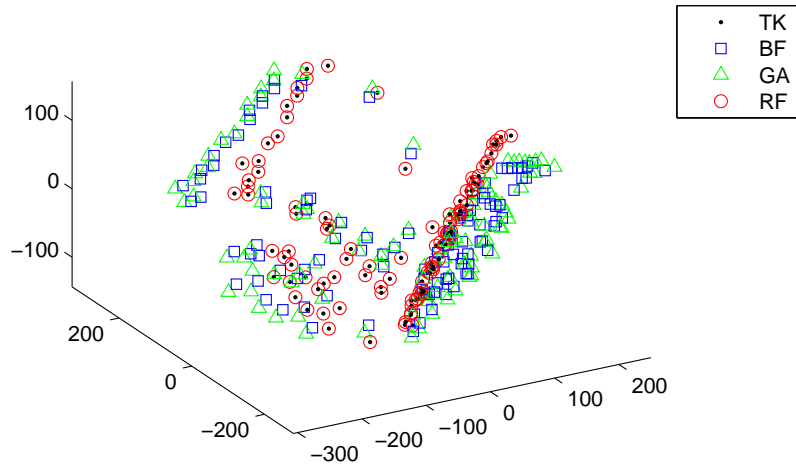


Figure 2.6: Reconstruction results of several methods by showing a top view of the hotel's 3D shape. Besides the reference shape (TK), we show the proposed method (RF) and two others (BF and GA).

Rank-based methods produce large errors because the degeneracy is not addressed explicitly. Since they assume data modeled as $\mathbf{W} = AB$ they try to estimate independently all values of A and B. In this way, they are estimating values c and f in equation (2.2), when these variables should be assigned by the orthogonality constraint (2.4).

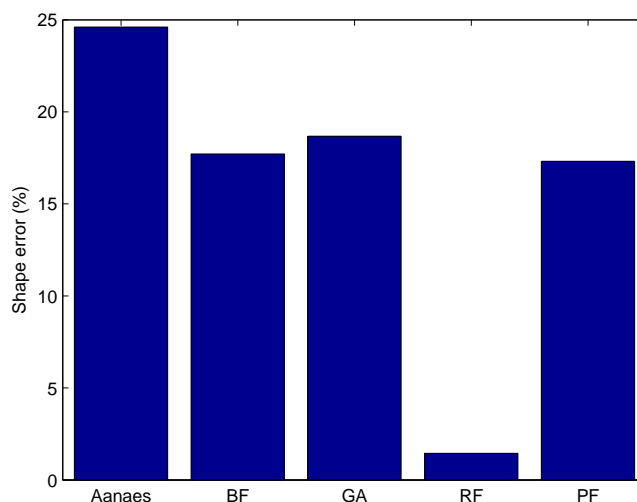


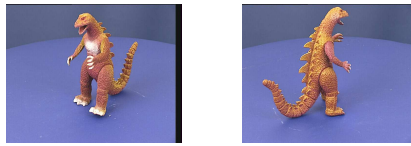
Figure 2.7: The relative error of several methods: the graph shows that none of the rank-based methods (Aanaes, BF, GA and PF) can compute shape adequately.

2.3.2 Dinosaur Experiment

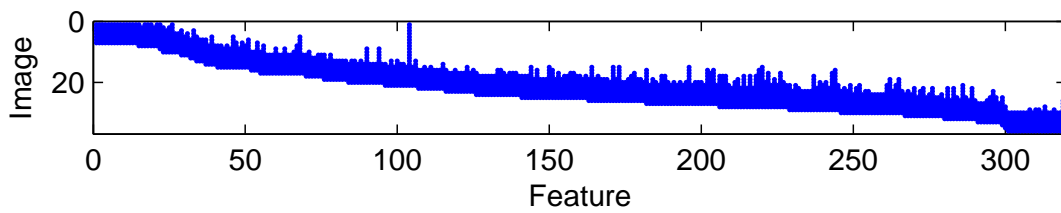
In the second experiment with real data, the proposed algorithm was tested with another well-known sequence - the dinosaur sequence⁴ (Figure 2.8). In the opposite direction from the hotel sequence, this one, which is composed by 36 images, does not contain any image with degenerate data and the measurement matrix is sparse with only 28% of known data (see Figure 2.8).

The dinosaur sequence was used to evaluate the performance of our “rigid factorization” algorithm with highly sparse data sets. Since the complete matrix is not available, we will compare our algorithm against the most accurate algorithm for this type of problem. To our knowledge, the Damped Newton algorithm of Buchanan&Fitzgibbon [Buchanan 05] produces the least reconstruction error (it is optimal for this criterion). The error measure is the root mean square error of the known data, that is $\epsilon = \|(W - MS) \odot D\|_2 / \sqrt{N_1}$. Here N_1 represents the number of observed points. In Figure 2.9 we show the position of the reprojection of the visible points ($w_i = m_i * S$) for one particular image of the sequence. The total Root Mean Square (RMS) error for the whole

⁴www.robots.ox.ac.uk/~abm



(a) Two images of the sequence



(b) Hotel 3D shape

Figure 2.8: Dinosaur’s sequence

sequence is 1.3705, though some points have a considerably high deviation. These points are shown in the right picture of Figure 2.9.

As both pictures show, except for peripheral areas, the majority of points are estimated with a reasonably low error. The fact that our algorithm exhibits larger error is an expected result due to the extra constraints imposed on the fitting model. Since we impose rigidity, there are less degrees of freedom to adjust to the error. For a qualitative evaluation, Figure 2.10 shows the dinosaur’s shape.

2.3.3 Full reconstruction with largely scaled images

In the third experiment, we present a real life example using the rigid factorization with missing data. The aim is to produce a 3D reconstruction of one building from a set of uncalibrated images.

We searched on Google for images depicting the building (search “casa da musica”) from a quite diverse set of viewpoints. The image scale is also quite diverse, since there are images taken from a pedestrian capturing one window together with far off aerial views from an airplane. Resolution

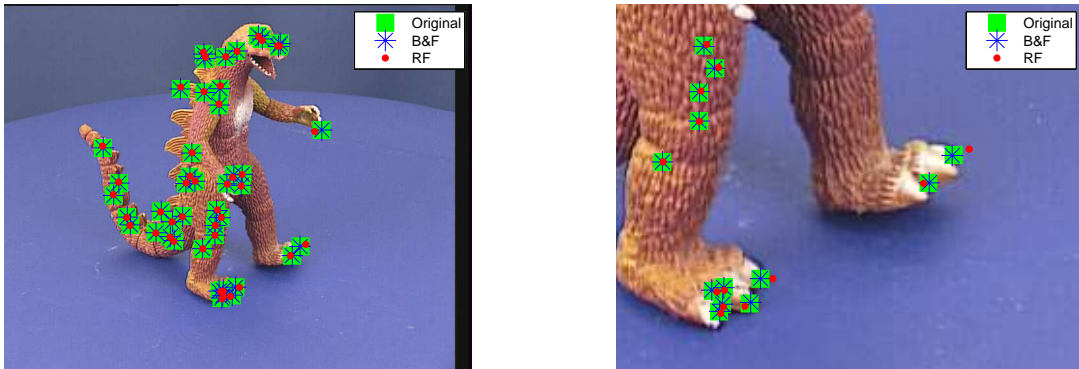


Figure 2.9: The left picture shows the original image points and their re-projections both for the proposed algorithm and the state-of-the-art. The RMS error is 1.3705 pixels, higher than the “best” value of 1.0847. Maximum error was 21 pixels, as the right figure shows.

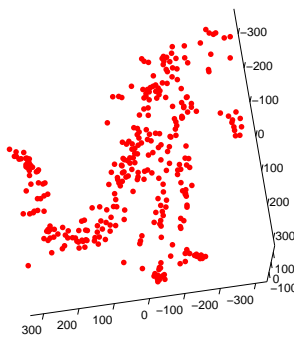


Figure 2.10: The dinosaur’s 3D shape

ranged from 3 Megapixel to a 60 Kilopixel and perspective effects were quite large in some of them. Image features were tracked by hand and were basically the vertices of the building and windows’ corners. In Figure 2.11 we show some of the 19 images with the features superimposed. Features are missing due to occlusion but also because we did not insert all visible points. Close up views with a large depth range produce strong perspective, therefore we just inserted points that were in a small depth range compared to the distance to the camera. Even though it favors the camera model adequacy it generates high volumes of (possibly degenerate) missing data.



Figure 2.11: An image sequence of Casa da Musica in Oporto - a “piecewise planar” building. Blue dots represent feature location.

A triangulation was computed to convey the shape in a more natural way. The blue surfaces are windows. As Figure 2.12 shows, the reconstruction is quite faithful. However perspective effects are noticeable, especially in the large front window. Nevertheless, in our opinion, this is quite a hard set of images for any 3D reconstruction algorithm and remember that no prior knowledge is used. For more precise applications this can be a good starting point for perspective factorization algorithms such as [Heyden 99, Sturm 96].

2.3.4 An Urban Modeling Example

In the last experiment, the scenario was also 3D building reconstruction (Figure 2.14). But in this case, the goal is very hard to achieve due to the few images available and to the high percentage of missing data. In this case, the object is composed of 21 features and there are 7 images available (Figure 2.13).

As the pattern of missing data in Figure 2.13 shows, the missing data is almost 60% and image 4 (bottom-right image in Figure 2.14) and image 5 are degenerate. Another relevant fact in this

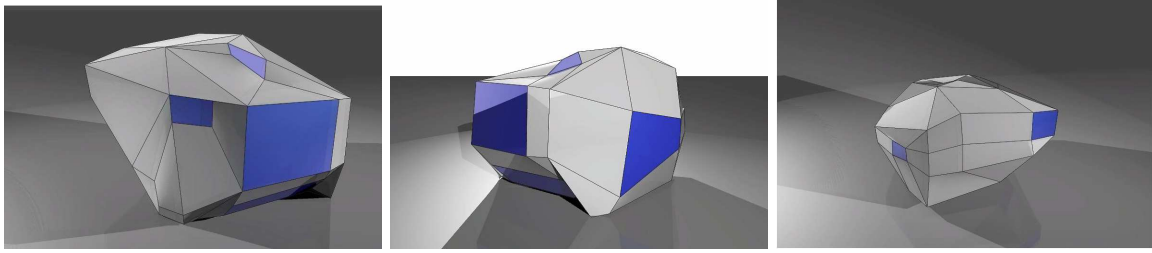


Figure 2.12: Reconstruction of Casa da Musica in Oporto

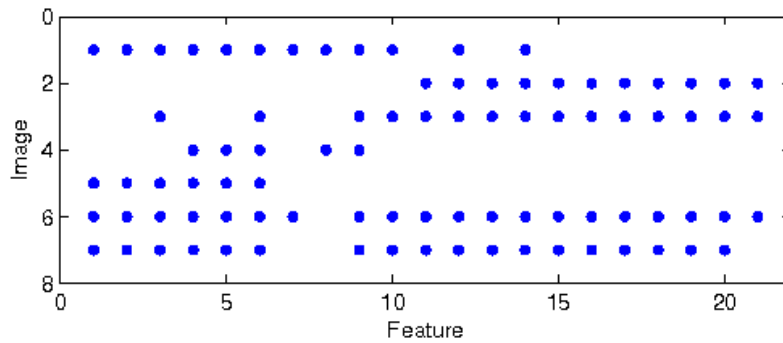


Figure 2.13: Pattern of missing points of the urban modeling

experiment is that features 7 and 8 only appear twice in the whole sequence (the minimum number) and feature 21 only 3 times.

Despite the difficulties mentioned above, we obtain quite a good result (Figure 2.15). In the top images of Figure 2.15, we can clearly see that the feature 8 is the only one in a wrong 3D position: it is not aligned with features 2 and 5 as it was expected (remember that feature point 8 appears 2 times only). This artifact disappears in the bottom images because the presented 3D reconstruction only uses the faces' corners of the building.

Through this example, we can verify that the presented method is a very useful tool for obtaining 3D reconstructions from few images of the considered object and a large percentage of missing data.

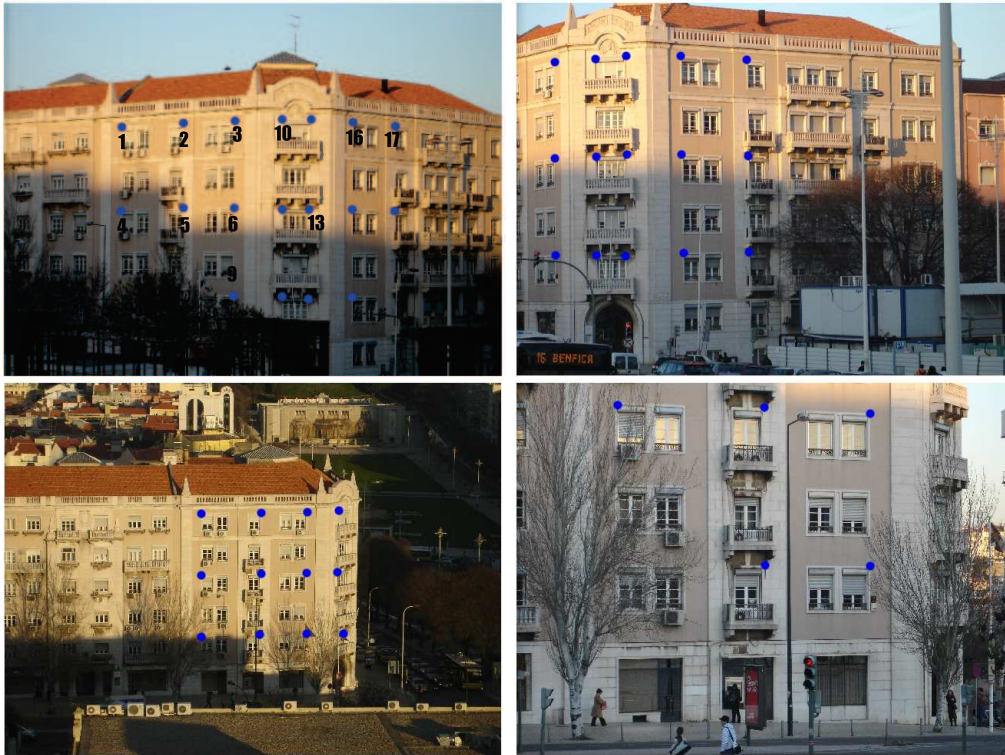


Figure 2.14: A typical example of urban modeling. From left-right and up-bottom, images #7-2-1-4 of the sequence. Top-left image has feature number superimposed.

2.4 Conclusions

In this chapter, we have presented a new factorization algorithm that computes the optimal motion shape and scale estimates (scaled orthography) from a feature track matrix. We also introduced an iterative algorithm that produces the same estimates when feature points are missing and the known points are degenerate, the most general and realistic situation.

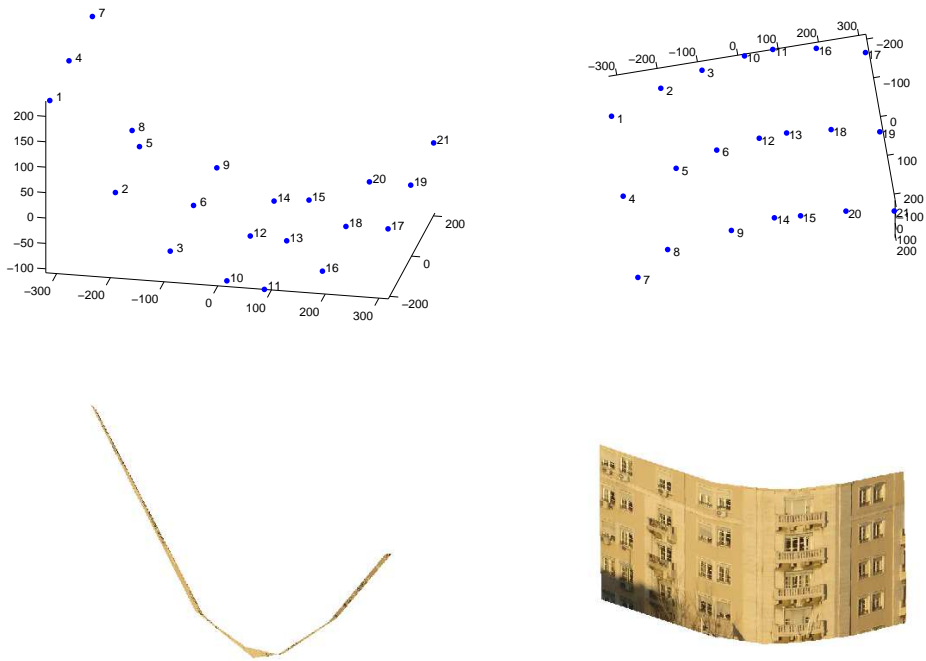


Figure 2.15: Top graphs show the 3D point reconstruction (shape). In the bottom figure, the image is mapped on a piecewise planar model, created with the corners of each plane.

Chapter 3

The Subspace Matching Theorem: Uniqueness Conditions for Correspondence between Point Clouds

The method for 3D reconstruction with missing data discussed in Chapter 2 assumed that points in several frames were put to correspondence. It required an ordered data matrix \mathbf{W} , such that each column represents the trajectory of one point in all frames and each row the coordinates of all points in one frame. Finding the correspondence between two images or one image and a 3D model is thus fundamental to scene understanding. Considering we observe a set of 2D sparse and unsorted points and known 2D/3D shape models, the main goal is to identify the object and compute its pose. In this chapter, we discuss the conceptual and theoretical aspects. The implementation issues, such as convergence rate and large-scale data, are topics of Chapter 4.

The scenario is defined by the following assumptions:

- We depart from a set of N points (2D or 3D) for which we must search for the correspondences.
- There are N points to be matched in both images (or 3D model) - the relation between the two sets is thus one-to-one.

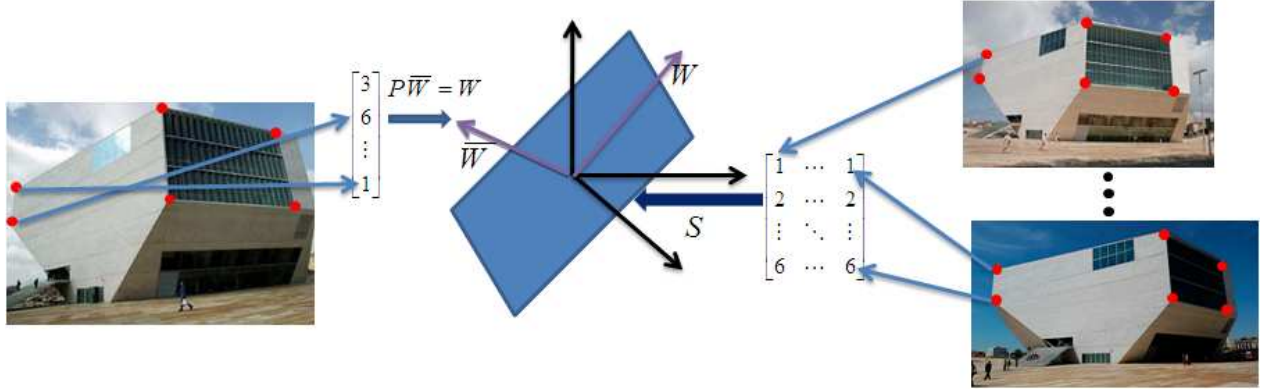


Figure 3.1: Subspace matching problem: Finding the correct permutation matrix \mathbf{P} such that the resorted observed feature vector $\overline{\mathbf{W}}\mathbf{P}$ lies on the subspace \mathbf{S} generated by the model. Note that the subspace can be computed from the 3D shape or from several image correspondences.

- We consider an affine camera model.

As we referred in the Introduction (Chapter 1), the point matching problem (Figure 1.4) can be casted by minimizing the “backprojection error function”

$$\begin{aligned}
 (\mathbf{M}^1, \dots, \mathbf{M}^F, \mathbf{P}^1, \dots, \mathbf{P}^F)^* = & \arg \min_{\mathbf{M}^1, \dots, \mathbf{M}^F, \mathbf{P}^1, \dots, \mathbf{P}^F} \sum_{f=1}^F \left\| \overline{\mathbf{W}}^f \mathbf{P}^f - \begin{bmatrix} \mathbf{M}^f & \mathbf{t}^f \end{bmatrix} \begin{bmatrix} \mathbf{S} \\ \mathbf{1}^T \end{bmatrix} \right\|_2^2 \\
 \text{s.t.} & \mathbf{P}^1, \mathbf{P}^2, \dots, \mathbf{P}^F \in \mathcal{P} \\
 & \mathbf{M}^f \in \mathcal{S}
 \end{aligned} \tag{3.1}$$

where \mathbf{M}^f and \mathbf{t}^f are the *camera matrix* a *translation vector* of frame f . The matrix $\overline{\mathbf{W}}^f$ is an unsorted version of the *measurement matrix* of frame f , \mathbf{W}^f , and \mathbf{P}^f is a permutation matrix, $\mathbf{W}^f = \overline{\mathbf{W}}^f \mathbf{P}^f$.

Solving optimization problem (3.1), we obtain the sorted projections and the *camera matrix* of F images given a known shape. Because model \mathbf{S} is a prior knowledge, the problem is decoupled in

F independent optimization subproblems, each one given by

$$\begin{aligned}
 (\mathbf{M}^f, \mathbf{P}^f)^* &= \arg \min_{\mathbf{M}^f, \mathbf{P}^f} \left\| \overline{\mathbf{W}}^f \mathbf{P}^f - \mathbf{M}^f \mathbf{S} \right\|_F^2 \\
 &\text{s.t.} \quad \mathbf{P}^f \in \mathcal{P} \\
 &\quad \mathbf{M}^f \in \mathcal{S}
 \end{aligned} \tag{3.2}$$

In the considered scenario, by centering the data the *translational vector* can be discarded. As it will be seen in Section 3.1, we propose an alternating scheme to find a solution for expression (3.2), splitting this optimization problem into two subproblems (w.r.t \mathbf{P}^f and \mathbf{M}^f). Finding permutation matrix \mathbf{P}^f hinges on a theoretical principle presented in Sections 3.2, 3.2.2 and Appendix D.

3.1 A Framework for Correspondence and Pose Estimation

Even though the methodology, presented in this section, is general for bilinear models, we will focus on designing solutions for two relevant computer vision problems: Image-to-image matching and image to 3D-model matching. In other words, we seek an efficient algorithm that is capable of:

- Putting to correspondence a large set of points in two images
- Putting to correspondence a large set of image points and its known 3D shape

In this chapter, we will provide sufficient conditions to solve the matching problems mentioned above, using convex programming and assuming an affine camera model. Simplifying the notation, (3.2) can be written as

$$\begin{aligned}
 (\mathbf{P}, \mathbf{M})^* &= \arg \min_{\mathbf{P}, \mathbf{M}} \left\| \overline{\mathbf{W}} \mathbf{P} - \mathbf{M} \mathbf{S} \right\|_F^2 \\
 \mathbf{P} &\text{s.t.} \quad \mathbf{P} \in \mathcal{P}
 \end{aligned}$$

Problem 4

where we replace $\overline{\mathbf{W}}^f$, \mathbf{P}^f and \mathbf{M}^f by $\overline{\mathbf{W}}$, \mathbf{P} and \mathbf{M} , respectively.

Invoking the definition of Frobenius norm, we have

$$\begin{aligned} (\mathbf{P}, \mathbf{M})^* = \arg \min_{\mathbf{P}} \quad & \text{trace} \left(\mathbf{P}^T \overline{\mathbf{W}}^T \overline{\mathbf{W}} \mathbf{P} - 2\mathbf{P}^T \overline{\mathbf{W}}^T \mathbf{M} \mathbf{S} + \mathbf{S}^T \mathbf{M}^T \mathbf{M} \mathbf{S} \right) \\ \text{s.t.} \quad & \mathbf{P} \in \mathcal{P} \end{aligned} \quad (3.3)$$

This is a mixed integer quadratic problem, very hard to solve. A common way of finding a solution for Problem 4 is reformulating it into two subproblems, resulting from either the camera or the assignment matrix is known:

- with known assignment, \mathbf{M} is computed by a pseudo-inverse
- with known camera matrix, the assignment \mathbf{P} can be computed efficiently using linear programming and/or branch and bound or the Hungarian method ([Veenman 01], [Maciel 03], [Fitzgibbon 03]).

Based on these two problems, an alternating scheme would be an intuitive way to solve Problem 4. If we have \mathbf{M}^k at iteration k , solving (3.3) in order to \mathbf{P} , the cost function is invariant to the first and third terms. The latter term is a constant. Since \mathbf{P} is a permutation matrix, we have $\mathbf{P}\mathbf{P}^T = \mathbf{I}$ and the first term is also a constant. Thus, knowing matrix \mathbf{M}^k , we obtain \mathbf{P} solving the following expression

$$\begin{aligned} \mathbf{P}^* = \arg \max_{\mathbf{P}} \quad & \text{trace} \left(\mathbf{P}^T \overline{\mathbf{W}}^T \mathbf{M}^k \mathbf{S} \right) \\ \text{s.t.} \quad & \mathbf{P} \in \mathcal{P} \end{aligned} \quad (3.4)$$

Note that (3.4) has an intuitive meaning: our goal is to find \mathbf{P} such that the projection of $\overline{\mathbf{W}}\mathbf{P}$ on the subspace spanned by $\mathbf{M}\mathbf{S}$ is maximum¹. With the new estimate \mathbf{P}^k , \mathbf{M}^{k+1} is the solution of a Least-Squares problem.

This cyclic coordinates approach is highly dependent of the initialization, due to the integer nature of the problem. There are few approaches in the literature that tackle Problem 4, the most

¹If \mathbf{M} is a rotation matrix, subspaces spanned by $\mathbf{M}\mathbf{S}$ and \mathbf{S} are equivalent.

notable one being the ICP algorithm [Besl 92]. Since the solutions provided by this methodology are not globally optimal and depending on the initialization, there is no guarantee that it may even reach a reasonable solution (e.g. strong rotation and scale in ICP). If the first assignment is poor, it leads to wrong motion solutions, feeding back error which becomes catastrophic in the next assignment. Likewise a wrong motion, most likely, will lead to poor assignments driving the process towards the same behavior. To circumvent this fact, we redefine our optimization criterion. Instead of using “back-projection error” to estimate the correspondence between two sets of points, we can obtain matrix \mathbf{P} from the following \mathbf{M} invariant equation

$$\overline{\mathbf{W}}\mathbf{P}\Pi_{\mathbf{S}}^{\perp} = \underline{\mathbf{0}} \quad (3.5)$$

where $\underline{\mathbf{0}}$ is a matrix of zeros and $\Pi_{\mathbf{S}}^{\perp}$ the projector into the orthogonal space to $\text{range}(\mathbf{S})$. The operator is given by matrix

$$\Pi_{\mathbf{S}}^{\perp} = \mathbf{I}_N - \mathbf{S}^T (\mathbf{S}\mathbf{S}^T)^{-1} \mathbf{S},$$

As we will see in next section, in noise-free conditions, Problem 4 and (3.5) have the same solutions.

3.2 Designing a Unique Solution for the Correspondence Problem

In noisy conditions, equation (3.5) will not be satisfied in general. The intuitive way of coping with this, is to define some criterion by which that equality is approximated, for example, in the least square error sense. Then a solution is found by solving the problem given by

$$\begin{aligned} \mathbf{P}^* = \arg \min_{\mathbf{P}} \quad & \|\overline{\mathbf{W}}\mathbf{P}\Pi_{\mathbf{S}}^{\perp}\|_F^2 \\ \text{s.t.} \quad & \mathbf{P} \in \mathcal{P} \end{aligned} \quad (3.6)$$

Taking into account that $\Pi_{\mathbf{S}}^{\perp} \Pi_{\mathbf{S}}^{\perp T} = \Pi_{\mathbf{S}}^{\perp}$ and replacing $\Pi_{\mathbf{S}}^{\perp}$ by its expression, the above cost function has the following form:

$$\left\| \overline{\mathbf{W}} \mathbf{P} \Pi_{\mathbf{S}}^{\perp} \right\|_F^2 = \text{trace} \left(\overline{\mathbf{W}} \mathbf{P} \mathbf{P}^T \overline{\mathbf{W}}^T - \overline{\mathbf{W}} \mathbf{P} \mathbf{S}^T (\mathbf{S} \mathbf{S}^T)^{-1} \mathbf{S} \mathbf{P}^T \overline{\mathbf{W}}^T \right) \quad (3.7)$$

$$= \text{trace} \left(\overline{\mathbf{W}} \mathbf{P} \mathbf{P}^T \overline{\mathbf{W}}^T - \overline{\mathbf{W}} \mathbf{P} \mathbf{S}^T (\mathbf{S} \mathbf{S}^T)^{-1} \mathbf{S} \mathbf{S}^T \mathbf{M}^T \right) \quad (3.8)$$

assuming that $\mathbf{P}^T \overline{\mathbf{W}}^T = \mathbf{S}^T \mathbf{M}^T$ (noiseless case).

Since \mathbf{P} is a full permutation, the first term of the cost function is constant and thus we obtain the following optimization problem,

$$\begin{aligned} \mathbf{P}^* = \arg \max_{\mathbf{P}} \quad & \text{trace} \left(\overline{\mathbf{W}} \mathbf{P} \mathbf{S}^T \mathbf{M}^T \right) \\ \text{s.t.} \quad & \mathbf{P} \in \mathcal{P} \end{aligned} \quad (3.9)$$

which is equal to (3.4). So, in noise-free conditions, Problem 4 is equivalent to (3.6). However both approaches suffer from the same drawback: the search space is not convex.

As referred before, \mathcal{P} is the set of permutation matrices. This set is defined as the set of matrices with elements 0 or 1, and which rows and columns sum to 1, that is:

$$\mathbf{P} \in \mathcal{P} \Leftrightarrow \sum_{i=1}^N \mathbf{P}_{ij} = 1 \quad (3.10)$$

$$\sum_{j=1}^N \mathbf{P}_{ij} = 1 \quad (3.11)$$

$$\mathbf{P}_{ij} \in \{0, 1\} \quad (3.12)$$

The combinatorial nature of the problem can be circumvented by relaxing the domain (\mathcal{P}), to its convex hull, the set of doubly stochastic matrices (\mathcal{D}_s). Being compact, this set is convex thus suggesting a better way to design efficient algorithms to seek the optimum. Mathematically, this set is obtained by replacing the non-convex constraint (3.12) in the above definition by the ‘‘convex’’ one $\mathbf{P}_{ij} \geq 0$. In simple words, we relax the $\{0,1\}$ constraint. The main problem here is that the solution is not guaranteed to be unique, and possible solutions may not even be permutations.

In fact, there are infinite solutions, resulting from the intersection of the polytope represented by equations (3.10,3.11) and $\mathbf{P}_{ij} \geq 0$, known as the Birkhoff polytope [Wolsey 99], and the linear subspace spanned by the shape matrix \mathbf{S} .

The surprising fact is that there exists conditions under which the above relaxation can be done, leading to a unique solution, that is to say, leading to the correct permutation. This fact converts what we thought to be a very hard problem into a clearly solvable and computationally simple one.

For clarity purposes we will be using current notation and focus on the particular case of 3D-2D matching. Considering the shape matrix $\mathbf{S} \in \mathbb{R}^{3 \times N}$ and the image points (observation matrix) $\overline{\mathbf{W}} \in \mathbb{R}^{2 \times N}$, we state the following

Theorem 1 *Subspace Matching Theorem* *If there are at least two known correspondences between 3D points and their 2D image projections, which for simplicity we define to be the first 2 coordinates, the solution of*

$$\overline{\mathbf{W}} \begin{bmatrix} I_{2 \times 2} & \mathbf{0} \\ \mathbf{0} & \mathbf{P}_0 \end{bmatrix} \Pi_{\mathbf{S}}^{\perp} = \mathbf{0} \quad (3.13)$$

with $\mathbf{P}_0 \in \mathcal{D}_s$, is unique and consequently \mathbf{P}_0 is a permutation matrix.

In general, if the model has dimension $N \times r$ and observations have dimension $N \times k$, the knowledge of $r - k + 1$ correspondences is required. We will introduce one simple “trick” by which one known correspondence can be dropped. For the considered case (3D-2D), we need to know only one correspondence. Alternatively, we can simply solve N equations such as (3.13).

Applying theorem 1, we solve the described matching problem through the following convex program instead of a high-complex integer problem (3.6).

$$\begin{aligned} \mathbf{P}^* = \underset{\mathbf{P}}{\operatorname{arg\,min}} \quad & \left\| \overline{\mathbf{W}} \begin{bmatrix} I_{r-k+1} & \mathbf{0} \\ \mathbf{0} & \mathbf{P} \end{bmatrix} \Pi_{\mathbf{S}}^{\perp} \right\|_F^2 \\ \text{Problem 5} \quad & \\ \text{s.t.} \quad & \forall i, j \sum_i \mathbf{P}_{ij} = 1, \sum_j \mathbf{P}_{ij} = 1 \\ & \mathbf{P}_{ij} \geq 0 \end{aligned}$$

Finally, remark that some degenerate cases exist, mostly inherent to the representation. In particular, if one row of \mathbf{S} is expressed by a convex linear combination of 2 others there will be a doubly stochastic row which also generates the image point. However, most notably in the noisy case, these degenerate cases do not affect the solution globally. In other words, if a small set of entries are degenerate, this only affects the corresponding elements of the permutation matrix.

3.2.1 A Computational Framework

In view of the above formulation, the correspondence problem can be solved by well-known optimization techniques. Taking the particular case of the ℓ_1 norm and recurring on the *epigraph technique* [Bertsekas 99] leads to the following linear program:

$$\begin{aligned}
 (\mathbf{p}, t)^* = \arg \min_t \quad & \sum_i t_i \\
 \text{s.t.} \quad & -t \leq \mathbf{G}\mathbf{p} \leq t \\
 & \mathbf{P} \in \mathcal{D}_s, t \geq 0
 \end{aligned} \tag{3.14}$$

where $\mathbf{G} = (\mathbf{S}^\perp \otimes \overline{\mathbf{W}})$, \mathbf{p} is $\text{vec}(\mathbf{P})^2$ and t is the vector of *slack variables*. Also the ℓ_∞ norm could be formalized in the same way, leading to an even smaller linear program. These formulations can be easily transposed to general purpose solvers or packages.

3.2.2 A Geometric Interpretation

Before we present a proof of the *Subspace Matching Theorem*, we will explain intuitively the main idea of Theorem 1 and why it can be used to find an integer solution through a convex set. Imposing the known points, we change the linear space spanned by the shape matrix (\mathbf{S}) into an affine subspace (revealed algebraically by the 2×2 identity matrix), as Figure 3.2 shows. Instead of a linear subspace which intersects the search space in non-vertex points, the intersection of such an affine subspace with the Birkhoff polytope is proved to be one vertex which is, by definition, a permutation matrix.

² $\text{vec}()$ stacks the columns of its argument into a single column.

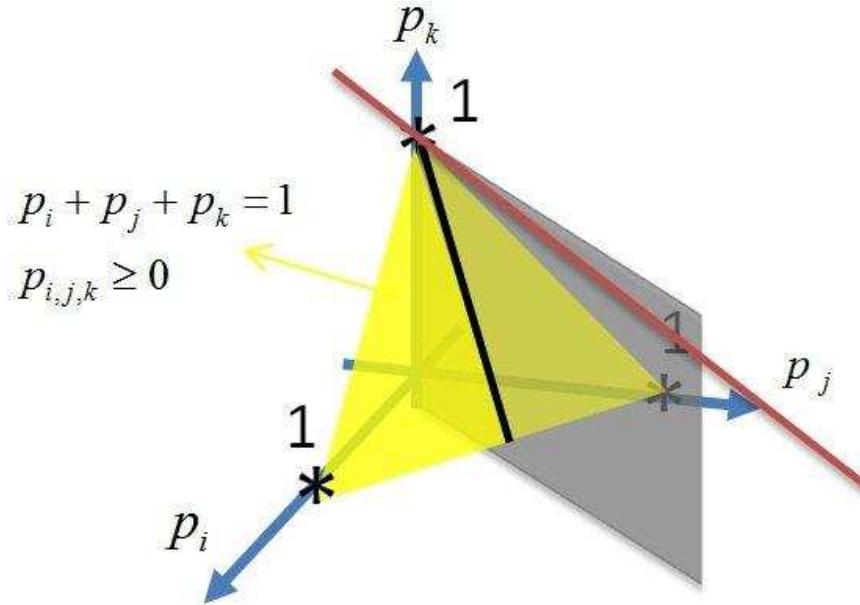


Figure 3.2: Subspace Matching: The yellow plane represents the convex-hull of permutation matrices, the gray one a linear subspace and the red line an affine subspace. The Subspace Matching theorem guarantees a unique solution because the affine subspace intersects the solution space (yellow plane) in one single point unlike the linear subspace, whose intersection with the solution is a line (a black one)

3.2.3 Subspace Matching Theorem: a Summary of the Proof

Here we will present a brief proof of the theorem, highlighting the 2D-3D case. The extended proof is in Appendix D.

Let N , r and k be positive integers, such that $r \geq k$, and $N \geq r + k$. Let

$$C = \begin{bmatrix} a^1 & a^2 & \dots & a^k \\ X^1 & X^2 & \dots & X^k \end{bmatrix} \in \mathbb{R}^{N \times k},$$

be a matrix where $a^i \in \mathbb{R}^{(r-k+1) \times 1}$ and $X^i \in \mathbb{R}^{(N-r+k-1) \times 1}$, $i = 1, \dots, k$ are given vectors. Also let S be a prescribed r -dimensional vector subspace of \mathbb{R}^N , such that the columns of C belong to S .

Then we have the following:

Theorem 2 *For generic matrix C and subspace S we have the following: if $M \in \mathbb{R}^{(N-r+k-1) \times (N-r+k-1)}$ is a doubly-stochastic matrix such that all vectors $\begin{bmatrix} a^i \\ MX^i \end{bmatrix}$, $i = 1, \dots, k$, belong to S , then M is the identity matrix.*

In the case we are interested in, we have $k = 2$ and $r = 2$, or $k = 2$ and $r = 3$, while N is large number. Here we just sketch the main ideas of the proof for $k = 2$ - the general case is done analogously.

Sketch of the proof for $k = 2$:

Our main goal is to find a vector v which is a linear combination of the vectors X^1 and X^2 such that

$$Mv = v. \tag{3.15}$$

Having this relation we use the doubly-stochastic property of M , to obtain strong restrictions on v . This relies heavily on the Perron-Frobenius theorem for nonnegative matrices.

First of all, there exists a permutation matrix P such that PMP^T has the form:

$$PMP^T = \begin{bmatrix} M_1 & 0 & \cdots & 0 \\ M_{21} & M_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ M_{l1} & M_{l2} & \cdots & M_l \end{bmatrix},$$

where M_1, \dots, M_l , are irreducible, i.e. cannot be further split in this way. Moreover, since M is a non-negative matrix, with row and column sums equal to 1, we have that all off-diagonal blocks are

equal to zero. So, from now on, we may assume that is in the block-diagonal form:

$$PMP^T = \begin{bmatrix} M_1 & 0 & \cdots & 0 \\ 0 & M_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & M_l \end{bmatrix} \quad (3.16)$$

with all blocks being irreducible.

Moreover, we have the following lemma:

Lemma 1 *If N is an irreducible doubly-stochastic matrix, and w is a vector such that $Nw = w$, then all entries of w are equal, i.e. there exists $c \in \mathbb{R}$, such that $w = c[1, \dots, 1]^T$.*

Proof of lemma:

Every row-stochastic matrix has as an eigenvector the vector $[1, \dots, 1]^T$, with the eigenvalue 1. On the other hand, by Geršgorin theorem (see below), all real eigenvalues of row-stochastic matrix are less than or equal to 1. Since N is an irreducible nonnegative matrix, by Perron-Frobenius theorem (see below), the multiplicity of its dominant eigenvalue (which as we have proved is 1), is equal to 1, i.e. it has only one linearly independent eigenvector corresponding to the eigenvalue 1. As we saw, this one is $[1, \dots, 1]^T$, as wanted.

Geršgorin theorem: Let $A = [a_{ij}] \in \mathbb{R}^{n \times n}$ be a square matrix. For every $i = 1, \dots, n$, denote by d_i the sum of absolute values of all non-diagonal entries of the i -th row, i.e.

$$d_i = \sum_{j \neq i} |a_{ij}|.$$

Then all (complex) eigenvalues of A lie in the union of discs with centers in a_{ii} with radius d_i , for all $i = 1, \dots, n$.

(Part of) Perron-Frobenius theorem: Let $A = [a_{ij}] \in \mathbb{R}^{n \times n}$ be a nonnegative square matrix, i.e. such that $a_{ij} \geq 0$, for every i and j . Then A has a real nonnegative eigenvalue λ corresponding to the eigenvector with all entries nonnegative, and such that all other eigenvalues μ of the matrix A , are such that $|\mu| \leq \lambda$.

If in addition A is irreducible, then λ has multiplicity 1.

Now, to find a vector v that satisfies (3.15), we do the following: Denote by Σ_i , $i = 1, 2$, the space of all vectors of the form $\begin{bmatrix} a^i \\ Y^i \end{bmatrix}$, where Y^i runs through $\mathbb{R}^{(N-r+1) \times 1}$, which belong to S . Since the space of all vectors $\begin{bmatrix} a^i \\ Y^i \end{bmatrix}$ has the dimension $N - r + 1$ and S has the dimension r , then their intersection, Σ_i , (in generic case) has dimension 3. Moreover, Σ_1 and Σ_2 are parallel lines, and hence they determine a plane Σ .

Denote by $p \subset \Sigma$ the line determined by $\begin{bmatrix} a^1 \\ X^1 \end{bmatrix}$ and $\begin{bmatrix} a^2 \\ X^2 \end{bmatrix}$, and by $q \subset \Sigma$ the line determined by $\begin{bmatrix} a^1 \\ MX^1 \end{bmatrix}$ and $\begin{bmatrix} a^2 \\ MX^2 \end{bmatrix}$. If they are parallel, then we have that $MX^2 - MX^1 = X^2 - X^1$, i.e. we can take $v = X^2 - X^1$.

Otherwise they intersect at some point T , and since Σ_1 and Σ_2 are parallel, we have that for some $\lambda \in \mathbb{R}$ we have the following:

$$T = (1 - \lambda) \begin{bmatrix} a^1 \\ X^1 \end{bmatrix} + \lambda \begin{bmatrix} a^2 \\ X^2 \end{bmatrix},$$

$$T = (1 - \lambda) \begin{bmatrix} a^1 \\ MX^1 \end{bmatrix} + \lambda \begin{bmatrix} a^2 \\ MX^2 \end{bmatrix},$$

and so in this case we have $Mv = v$, for $v = (1 - \lambda)X^1 + \lambda X^2$.

Finally, with the obtained tools, we can prove that $M = I_{N-r+1}$. First of all, put M in the block-diagonal form (3.16), with all blocks M_i being irreducible of size d_i . If all d_i are equal to 1, we are done. Otherwise, split the vectors PX^1 , PX^2 and Pv in the blocks of the corresponding dimensions d_i : $PX^i = [x_1^i, \dots, x_l^i]^T$, $Pv = [v_1, \dots, v_l]^T$. Then we have that for all $i = 1, \dots, l$:

$$M_i v_i = v_i,$$

and so by Lemma 1 there exists $c_i \in \mathbb{R}$ such that $v_i = c_i [1, \dots, 1]^T \in \mathbb{R}^{1 \times d_i}$. Thus we would have that the vectors x_i^1 , x_i^2 and $[1, \dots, 1]^T$ are linearly dependent, which generically is not satisfied.

Thus $M = I$, as wanted.

3.3 Summary

This chapter tackled the problem of finding correspondences between feature vectors when prior knowledge existed about their geometric properties. In particular, if data is constrained to a linear subspace, feature matching is the process by which the feature vector entries are sorted such that it lies on the known subspace. Under noisy conditions, this is a very hard combinatorial problem to solve.

In this chapter, we proved that the original integer problem has a unique solution in a convex set. This theoretical result has great importance because the matching problem, typically of a combinatorial nature, can be solved by convex optimization tools, namely, linear programming.

However as we will see in Chapter 4, this implementation is not the most efficient because its structure explodes in size, in particular the constraint matrix \mathbf{G} . In conclusion, specially designed algorithms are in order here. This issue is the subject of the next chapter.

Chapter 4

Designing Optimal Search Algorithms for Large Scale Point Correspondence

In this chapter, we will show a global framework for finding the correspondence between models and data in a real scenario (in the presence of outliers and clutter points). This new framework allows us to incorporate, in a natural way, local and diverse information with geometric (global) constraints which is imposed by Theorem 1 and presented in Chapter 3. Related with Computer Vision challenges, our methodology solves, in a global way, the 2D-2D, 3D-3D and a more difficult problem: the correspondence between 3D model and its 2D projections. In these cases, the method combines geometric constraints with local feature descriptors, like photometry.

4.1 State-of-the-Art

Looking at the aforementioned subjects, several algorithms have been used to solve the object recognition problem from single images - feature/appearance and geometric approaches. Unlike the 2D-2D recognition problem [Rodrigues 08], a common strategy to recognize a 3D object from one single 2D image is solving a matching problem between two points clouds. Successful methodologies in point matching tend to place their strategies in two main classes of algorithms:

- those that rely on very robust local image descriptors which turn the matching stage much simpler (e.g. Nearest Neighbor)
- those algorithms that solve efficiently a class of combinatorial problems but with a limited scope due to strong modeling (assumptions)

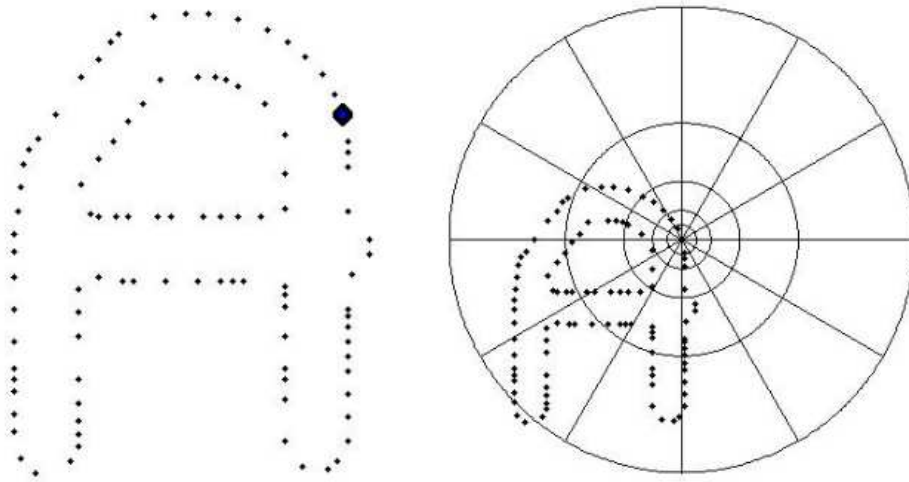


Figure 4.1: *Shape Context*, an example of 2D-2D shape matching approach. This method uses a log-polar histogram to build up a shape descriptor that measures similarity between 2D shapes. The matching is solved through a linear assignment problem.

In the first case, there is an extensive list starting with David Lowe’s SIFT [Lowe 99], *Shape context* by Belongie et al. [Belongie 02] or [Berg 05], where feature descriptors and geometric constraints (like proximity) are combined with optimization strategies that lead to a somehow “mixed” scheme. Usually geometric information (rigidity, homography) is used to validate the matching, which is inherently suboptimal. In [Belongie 02], the *shape context* descriptors are used to compute the two sets correspondence, using linear programming. These local descriptors attach to each point a histogram of the orientation and distance between itself and all other points. The SIFT algorithm [Lowe 99] gives a feature vector for each image 2D point, based on the Gaussian filters’ output at these 2D points. In the first stage, this strategy selects automatically the candidate points to be matched and the correspondence is performed by minimizing the distance between feature vectors of points in different images. Based on the SIFT descriptors, several methods were

proposed recently, such as [Morel 09]. The work presented in [Berg 05] allows us to compute the match between two 2D views of a 3D object using appearance information (feature descriptors) and geometric constraints, e.g. proximity between points. Smooth motion is required and the algorithm's convergence is not guaranteed because the solution is found by solving a linearized version of an integer quadratic problem.

Instead of the referred methods above, the second class of algorithms lies on geometric and global constraints. The work presented in [Lowe 87] recognizes a 3D object from a single 2D image by making line-to-line and point-to-point correspondences. To compute the solution (camera parameters and all features matches), this method needs to know *a priori* a set of correspondences. In [Torresani 08], the authors build a cost function based on geometric and appearance information and find the solution through graph-cuts. To compute the correspondences between two 2D and 3D sets, [Gold 98] proposes a highly non-convex optimization problem, which is solved using deterministic annealing. This approach requires a rough alignment between the two sets. The matching between two sets of 3D points is estimated by ICP-like algorithms [Besl 92, Fitzgibbon 03]. In [Fitzgibbon 03] the 2D case is also discussed and the Levenberg-Marquardt algorithm is used to find the solution. An EM-type algorithm is proposed in [David 04], but like [Besl 92, Fitzgibbon 03], the convergence is not guaranteed and some solutions can correspond to local *minima*. A quadratic optimization problem is suggested in [Pires 08] to find the 2D-2D matching solution. Such as [Lowe 87], the authors of [Pires 08] use a Newton-based algorithm to minimize the cost function, but a global minimum may not be found. A global solution for the 3D-3D correspondence can be found through a *branch-and-bound* algorithm [Li 07], a similar strategy used in [Breuel 03, Olsson 08]. In [Olsson 08], the proposed RANSAC-based approach is a computationally expensive method. Imposing rank constraints through a linear program [Oliveira 05] is another approach to reach the global optimum. However, similar to some mentioned methods, the solution given by the algorithm depends critically on the initial estimate. A linearized version of a non-convex problem can be built by a successive convexification process and allows us to compute the matching solution using linear programming [Jiang 07], but a smooth camera motion is required.

Another way to obtain the match is evaluating the eigenvalues spectrum of an affinity matrix [Leordeanu 05], by solving linear programs. As in descriptor based approaches, *a priori* information about the similarity between each pair of features is required.

Our approach belongs definitely to the second class of methods but, in our view, we follow a more general framework. We model the assignment problem in the natural way - a 0-1 optimization problem - but we developed slightly different reformulations allowing convex relaxations that lead to “easy” problems with the same exact minimizer.

One may argue that most of the matching and registration systems have in their intimate core the two simplest algorithms: ICP [Besl 92] and RANSAC [Fischler 81]. The former is quite efficient computationally but very unreliable when two clouds of points are far off registration, becoming highly dependent on the initialization. The latter is highly robust to most settings but can be highly inefficient, especially if used in a point matching context.

Our framework is easily adapted to recent projected gradient algorithms (Nesterov [Nesterov 03], Barzilai-Bowein [Barzilai 88]) which provably converge, sometimes with a superlinear convergence rate. These algorithms can cope with millions of variables and are quite insensitive to the initialization. Therefore, the methodology presented in the paper allows us to obtain solutions that are both robust to initial conditions and computationally efficient. Though the theoretical core guarantees the correct solution for the 1-to-1 correspondence problems only, such as [Li 07, Wang 09], we developed an optimization procedure nicknamed “Guided Search Consensus (GUISAC)” due to the resemblance to the RANSAC algorithm. The roots of GUISAC approach are on the experimental evidence that a small set of points converges much faster towards the correct match. Due to the intrinsic characteristics of the matching criterion, this strategy speeds up significantly the optimization algorithm’s convergence and is able to deal with outliers and clutter points.

4.2 A Large-Scale Formulation

In Chapter 3, we have shown a theoretical result enabling the formulation of the matching problem between two point clouds as a convex problem. In this section, we reformulate Problem 5 in

order to deal with a large number of variables (the number of variables are N^2 , where N is the number of points). Several minimization methods can satisfy this requirement but a particular type of algorithm, the projected gradient algorithms, deals with a large amount of data featuring low computational cost and high convergence rates.

Though the projected gradient methods are applicable to a generic convex program, this strategy performs better if the estimation of the points projection on the constraints set is computationally simple. In our case, the natural constraint set is \mathcal{D}_s , the set of doubly-stochastic matrices. However, there is no closed-form solution for projecting a matrix in this set. Devising an algorithm to finding the closest doubly-stochastic matrix to a given matrix has a similar degree of complexity to solving Problem 5.

In order to promote all capabilities of the projected gradient methods, we must redraw our optimization scheme, replacing Problem 5 with

Problem 6

$$\mathbf{P}^* = \underset{\mathbf{P}}{\operatorname{arg\,min}} \left\| \overline{\mathbf{W}} \begin{bmatrix} I_{r-k+1} & \mathbf{0} \\ \mathbf{0} & \mathbf{P} \end{bmatrix} \Pi_{\mathbf{S}}^\perp \right\|_F^2 + \frac{c}{2} \|\mathbf{P}\mathbf{1} - \mathbf{1}\|_F^2 + \frac{c}{2} \|\mathbf{P}^T\mathbf{1} - \mathbf{1}\|_F^2$$

s.t. $\mathbf{P}_{ij} \geq 0$

where $\mathbf{1}$ is a column vector of 1's. Moving the equality constraints to the cost function as a penalty leads to a problem that can easily be framed in any projected gradient method: Projecting a matrix into the set of matrices with non-negative entries boils down to replacing negative entries with 0, that is $\operatorname{Proj}(X) = \{\forall i, j \ X_{ij} = \max(X_{ij}, 0)\}$. In the noiseless case Problems 5 and 6 are equivalent. In noisy conditions, the doubly stochastic solution is attained by duality theory. The dual problem has a similar structure to Problem 6 but with additional complexity without increased precision in all practical situations considered.

Being an efficient implementation of the theoretical core, Problem 6 greatly increases the applicability range of *Subspace Matching theorem* (Chapter 3 and Appendix D). The next section proposes additional constraints to deal with more difficult scenarios like outliers and clutter points, keeping the same computational complexity. Note that these new features are not covered by the

theorem. Unlike Problem 6, the optimization programs listed below have an infinite number of solutions, if we consider solely the rigidity constraint.

4.2.1 Clutter Points


$$\mathbf{W} = \begin{bmatrix} \mathbf{u}_1 & \mathbf{v}_1 \\ \mathbf{u}_2 & \mathbf{v}_2 \\ \mathbf{u}_3 & \mathbf{v}_3 \\ \mathbf{u}_4 & \mathbf{v}_4 \end{bmatrix} \quad \mathbf{P} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad \mathbf{PW} = \begin{bmatrix} \mathbf{u}_1 & \mathbf{v}_1 \\ \mathbf{u}_2 & \mathbf{v}_2 \\ \mathbf{u}_3 & \mathbf{v}_3 \\ \mathbf{u}_4 & \mathbf{v}_4 \end{bmatrix}$$


Figure 4.2: Correspondences with elimination: In this case, the wide partial permutation matrix \mathbf{P} has one column of zeros. Each full column with zeros eliminates one row from the data matrix \mathbf{W} . The remaining columns perform the required permutation.

So far, the presented theorem and its implementation solely admits the same number of points in the two used sets: \mathbf{S} and $\overline{\mathbf{W}}$. However, in a more realistic scenario, these numbers can be different.

Consider a very common situation: image $\overline{\mathbf{W}}$ has M 2D points and the 3D model is composed by N points, $M > N$. This difference $M - N$ is the number of clutter points. Our goal is to estimate the matching between image and model points, discarding these clutter points. We have to redefine Problem 6 to cope with the difference in dimensions.

A natural way to extend the problem is considering \mathbf{P} as partial doubly-stochastic \mathcal{D}_s^{M-N} , $\{\mathbf{P} : \sum_i \mathbf{P}_{ij} \leq 1, \sum_j \mathbf{P}_{ij} = 1, \mathbf{P}_{ij} > 0\}$. In other words, our variable \mathbf{P} is a wide matrix, instead of a square matrix, where the columns corresponding to clutter point must sum to zero, as Figure 4.2 shows. However, following this approach, we add an unnecessary complexity to the optimization process because the inequality constraints $\sum_i \mathbf{P}_{ij} \leq 1$.

To keep each iteration as simple as possible, we define the following problem

$$\begin{aligned} \mathbf{P}^* = \arg \min_{\mathbf{P}} & \left\| \overline{\mathbf{W}} \begin{bmatrix} I_{r-k+1} & \underline{\mathbf{0}} \\ \underline{\mathbf{0}} & \mathbf{P} \end{bmatrix} \begin{bmatrix} \Pi_{\mathbf{S}}^\perp \\ \underline{\mathbf{0}} \end{bmatrix} \right\|_F^2 + \frac{\epsilon}{2} \|\mathbf{P}\mathbf{1} - \mathbf{1}\|_F^2 + \frac{\epsilon}{2} \|\mathbf{P}^T\mathbf{1} - \mathbf{1}\|_F^2 \\ \text{s.t.} & \quad \mathbf{P}_{ij} \geq 0 \end{aligned} \quad (4.1)$$

Such as Problem 6, variable \mathbf{P} is also a square matrix with dimensions $M \times M$ (instead of $N \times N$). The image points selected as clutter ($M - N$ columns of \mathbf{P}) do not influence the value of the cost function because the null space was extended with zeros.

4.2.2 Adding *a Priori* Knowledge

The rigidity assumption about the world is a general constraint that enables solving the matching problem without any prior information. However, in some real applications (i. e. tracking), local properties, such as maximum displacement or feature descriptor similarity may help increasing precision, speed and reliability. Our approach can naturally incorporate this local information by adding an extra (linear or quadratic) term to the cost function of Problem 5, leading to the general problem

Problem 7

$$\begin{aligned} \mathbf{P}^* = \arg \min_{\mathbf{P}} & \left\| \overline{\mathbf{W}} \begin{bmatrix} I_{r-k+1} & \underline{\mathbf{0}} \\ \underline{\mathbf{0}} & \mathbf{P} \end{bmatrix} \begin{bmatrix} \Pi_{\mathbf{S}}^\perp \\ \underline{\mathbf{0}} \end{bmatrix} \right\|_F^2 + \frac{\epsilon}{2} \|\mathbf{P}\mathbf{1} - \mathbf{1}\|_F^2 + \frac{\epsilon}{2} \|\mathbf{P}^T\mathbf{1} - \mathbf{1}\|_F^2 + \alpha \mathbf{1}^T (\mathcal{C} \odot \mathbf{P}) \mathbf{1} \\ \text{s.t.} & \quad \mathbf{P}_{ij} \geq 0 \end{aligned}$$

where α is a penalty constant. Each element of matrix \mathcal{C} (c_{ij}) represents the cost of assigning point i in the image to the model's point j (3D model), based on local feature information, e.g. color, brightness, feature descriptors, such as Figure 4.3. Values c_{ij} can be distances on the feature descriptor's manifold (SIFT, cross-correlation, distance of histograms). Thus, when two points have the same local descriptor we have $c_{ij} = 0$. If the local information implies discarding some

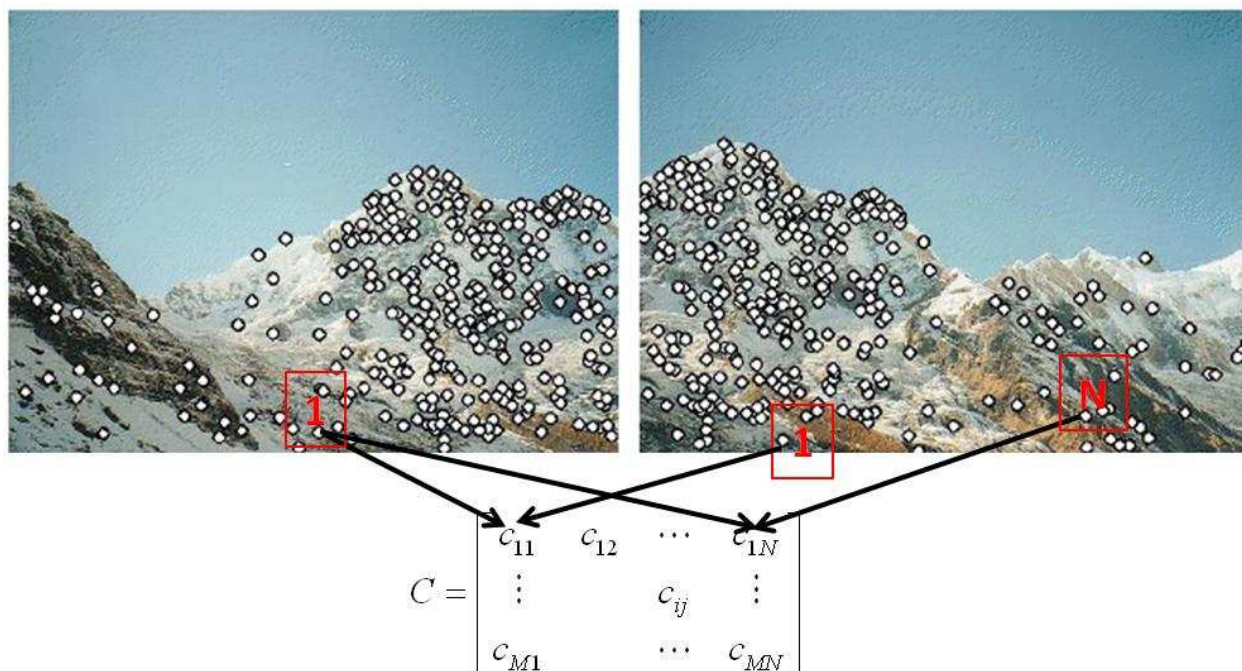


Figure 4.3: Including non-geometric costs - Using local information, each element of matrix C may express the dissimilarity between any two points: brightness, feature, color, displacement smoothness. Also, support constraints like *maximum disparity allowed* can be easily coded with this term by setting $C_{ij} = \{0, \infty\}$.

assignments ($c_{ij} = \infty$), thus making C a support matrix (imposes some $p_{ij} = 0$). This hard *a priori* decision cuts the search space heavily. Spatial penalties like maximum disparity can also be coded by this framework. It is important to note that this new term does not increase the computational complexity of the minimization algorithm (see Appendix D).

4.3 Guided Search Consensus

After discussing the method's extensions to deal with clutter points and prior knowledge in Sections 4.2.1 and 4.2.2, we will address a cross-cutting issue to any iterative algorithm: the convergence rate. Based on empirical evidence, we propose a strategy to achieve a more computationally efficient search.

The speed of convergence may be dramatically increased if we explore specific knowledge. As a

consequence of the models explained earlier, object pose can be computed if at least 4 correspondences are known. Thus, using 4 hypothetical matches, the camera model can be instantiated and the 3D object shape backprojected to the image. Using the simplest matching heuristic (Closest Point), a new set of correspondences can be computed and the model checking performed to measure the quality of the fit. This is the general framework of RANSAC where the hypothetical matches have to be selected randomly. Exploiting the fact that we have a convex problem, and that the convergence is smooth towards an integer minimizer - a permutation matrix -, we devised a guided search strategy, called GUI SAC, with a model checking step, described in Algorithm 4.3.

Algorithm 4.3: Guided Search Consensus (GUI SAC)

Searching for $\mathbf{P}^* = \operatorname{argmin}\{f(\mathbf{P})\}, \text{ s.t. } p_{ij} \geq 0$

1 - Initialize \mathbf{P} , $k = 0$

While $ErrorCriteria(\mathbf{P}_k) > Thresh$

2 - Compute \mathbf{P}_{k+1} performing an iteration of projected gradient method

3 - If at least 4 "reliable" matches exist

Compute pose \mathbf{M} and project model onto the image, \mathbf{MS}

Using NearestNeighbours get new match \mathbf{P}_{NN}

4- If $f(P_{NN}) < f(P_{k+1})$

then $P_{k+1} = P_{NN}$

end

end

$k = k + 1$

end

This procedure naturally fits any optimization algorithm.

In order to show the speed improvement, we compare GUI SAC with two other variations of this methodology, namely

- Projected Gradient: Pure gradient algorithm, iterating until the error is below a chosen threshold.
- PG w/ Projection: Projected Gradient with Projection is exactly like Algorithm 4.3 but doing projection on the set of Permutation matrices instead of choosing 4 points as GUI SAC suggests. Every other m iteration, the closest permutation matrix to the current estimate is

calculated using the Hungarian method. This new point is admitted only if the cost function value is smaller than the previous one.

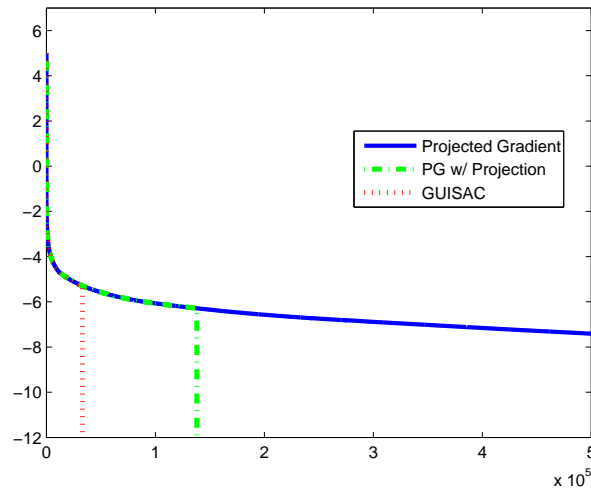
Both strategies and GUI SAC use Nesterov’s projected gradient as minimization algorithm. The main challenge is in computing the parameters needed to implement the projected gradient algorithm (see Appendix E).

As Figure 4.4a shows, by exploring the intrinsic characteristics of the matching problem, we are able to find the correct solution much faster than a gradient algorithm. Comparing the two truncation methods mentioned before, we verify that the proposed algorithm is computationally cheaper than a projected gradient with projection on the set of permutation matrices. Since the minimization algorithm tends to establish correct correspondences for some points much sooner than the rest, computing an affine transformation with a subset of points substantially increases the convergence velocity. The error curves, shown in Figure 4.4a, were obtained from one example with 100 points.

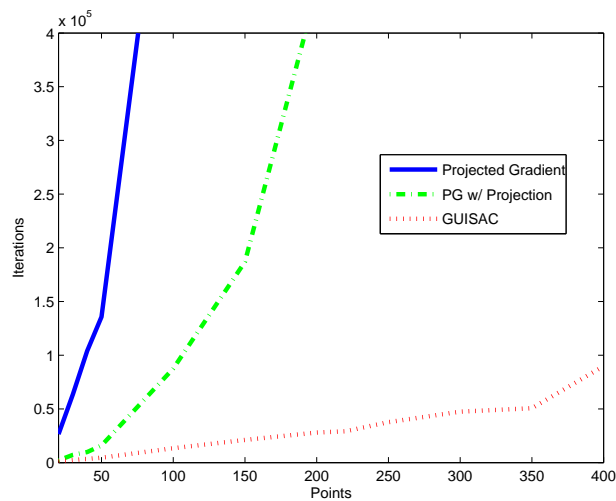
We performed a set of random experiments with increasing dimensionality to test how the three variations scale up with the number of points. Figure 4.4b highlights the virtues of GUI SAC: its complexity scales up very smoothly, unlike the other two strategies.

4.4 Experimental Results

To evaluate and measure the response of our method under noisy conditions, we will present several synthetic and real experiments. The first type of experiment is done with synthetic data and will allow us to measure the algorithm’s robustness to noise. The clear advantage of its global nature is highlighted when comparing it against a local approach [Sudhir 97].



(a) (log) Error Vs Number of iterations for one single experiment matching 50 points.



(b) Iterations Vs. dimensionality:
Computational complexity of all 3 approaches with problem size

Figure 4.4: Evaluation of the GUISAC convergence speed against two different approaches. Notice the sharp downward slope of the few initial iterations vs the slow convergence towards the optimum. GUISAC leverages on the former advantage and avoids the latter inconvenient. Graph (b) shows the very low increase in computational load as the problem scales up. Complexity stays low and for more than 200 points, other algorithms are not applicable.

In the second part of this section, several 2D-2D and 2D-3D real experiments will be shown. Some experiments are very challenging to any correspondence algorithm due to repetitive patterns in the images of the object models. We use real data also to evaluate the resilience of GUI SAC to clutter and outliers.

In order to better capture the algorithm's behavior, we use two error measures:

1. The projection error: the error between original data \mathbf{W}_{orig} and estimated data $\mathbf{P}^*\overline{\mathbf{W}}$, $\frac{\|\mathbf{W}_{orig}-\mathbf{P}^*\overline{\mathbf{W}}\|}{N}$.
2. The number of wrong matches $\frac{\|\mathbf{P}-\mathbf{P}^*\|}{2}$. This measure counts the number of 1's off their correct position in the estimated permutation matrix¹

For both type of errors, we compute the *reference error* which represents the concept of ground-truth. The *reference* projection error uses the original permutation matrix \mathbf{P} used in the error function, instead of \mathbf{P}^* . For the second error measure, we compute the optimal assignment between the noise-free (\mathbf{W}_{orig}) and the noisy data matrices (\mathbf{W}). This measure sets the reference number of wrong matches.

4.4.1 Synthetic data

Two type of experiments were performed to measure the algorithm's robustness to noise. In the first experiment, we defined 9 different noise levels, and for each of them, 200 independent experiments were run. The 3D object is composed of 40 randomly generated points. In each experiment, we used two different sets: a 3D object (\mathbf{S}) and one 2D image ($\overline{\mathbf{W}}$), generated by an orthographic projection with added gaussian noise ($\mathcal{N}(0, \sigma^2)$).

In this first experiment, we compare two possible implementations of our approach which differ on the final computation of permutation matrix.

- Our approach 1: The estimated permutation results from the minimum error between all estimated points $\overline{\mathbf{W}}\mathbf{P}^*$ and all original points \mathbf{W}_{orig} . It is computed using the Hungarian

¹Note that each wrong match counts as two 1's: the wrong position of the 1 and the 0 left in the correct position

method.

- Our approach 2: The estimated permutation is calculated from an affine transformation computed using the n “best” matches.²

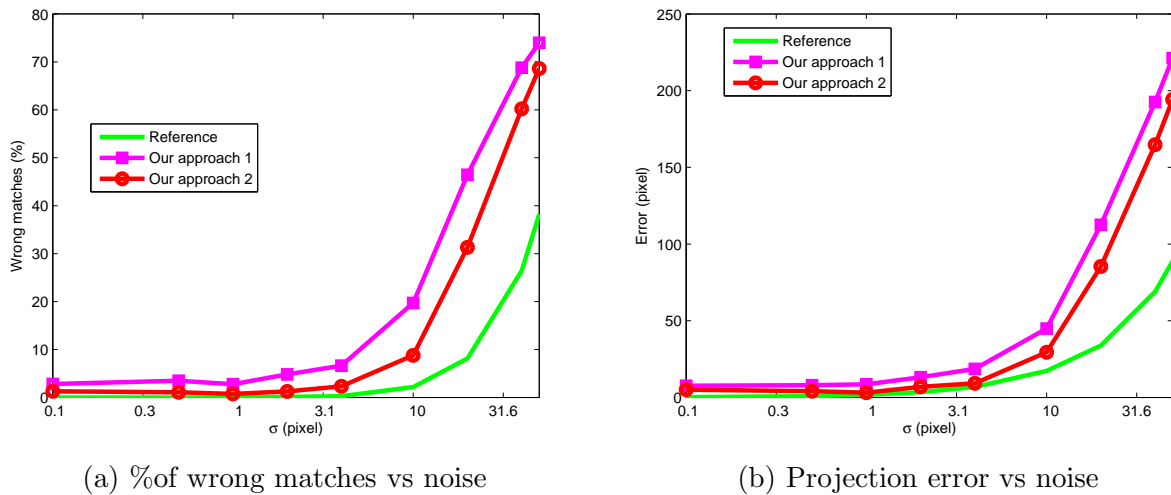


Figure 4.5: Comparing two matching approaches. Note that for $\sigma > 5$ pixels the image is so distorted that the shape matching paradigm is meaningless.

In Figure 4.5, we show the noise standard deviation in pixels whereas image size is 1000×1000 . If $\sigma = 3$ pixels ($3\sigma = 9$) the percentage of wrong matches is clearly below 5% (Figure 4.5a) and the projection error is negligible comparing with the *reference* solution as Figure 4.5 illustrates. For higher noise levels (ex. $\sigma = 10$), we obtained 90% of correct matches. Note that with such level of noise, 1% of the points may have up to 100 pixels deviation in an image of size 1000×1000 .

According to the pictures of Figure 4.5, we obtained the best performance with the second approach. This strategy is similar to GUIAC: when the found solution is not the totally correct one, we can improve the estimative using the n “best” matches. Due to the best performance, we use the second approach from now on.

As Figure 4.6 shows, we benchmark our method (red dots) against the algorithm proposed in [Sudhir 97] (SZB) (blue triangles), performing 200 independent experiments per each 9 different

²In the presented experiments we use $n = 5$.

noise levels. The 3D object is also composed of 40 randomly generated points. Comparing this experiment with the former, the main difference is a higher perturbation ($\mathcal{N}(0, 20\sigma^2)$) added to one (solid line) or two (dashed-line) projections. Due to the difference of the noise's power, the last points can be classified as outliers. It is important to refer that the affine transformation is computed using 4 correspondences [Sudhir 97] (blue triangles) and one of them is the outlier point.

In Figure 4.6, we show the noise standard deviation in pixels whereas image size is 1000×1000 . Considered the range of noise between 0.1 and 50 pixels, our algorithm achieves a better performance than the algorithm suggested in [Sudhir 97]. Because the proposed method acts globally, the influence of outliers has less impact instead of SZB where one bad choice can jeopardize the search.

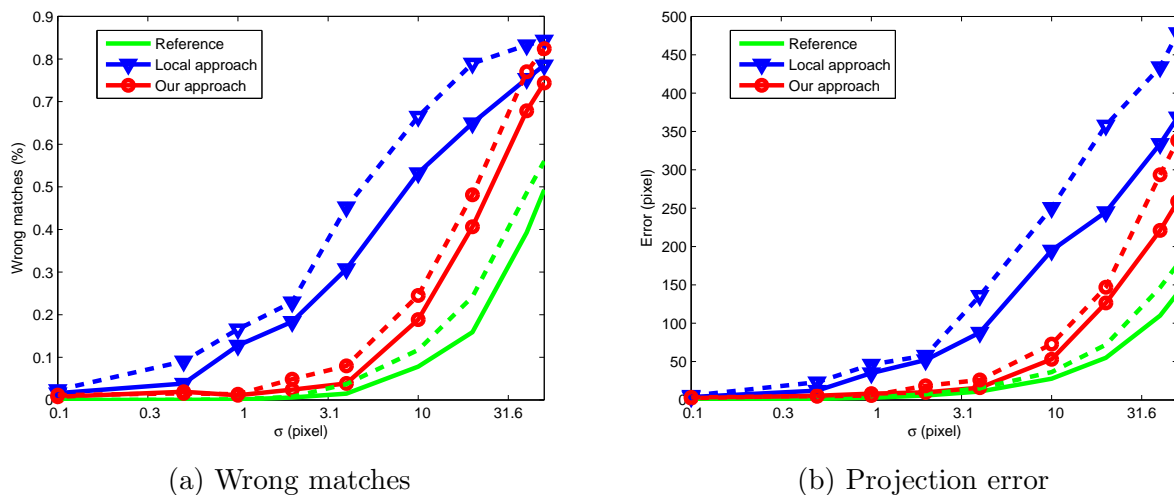


Figure 4.6: Comparing the proposed global method against a local approach: our method (red dots) Vs. SZB[Sudhir 97] (blue triangles). Solid line contains one outlier point in the data set and dashed line two outlier points. Because the proposed approach acts globally the influence of these points has less impact unlike the local strategy SZB.

Comparing the graphs of Figures 4.5 and 4.6, we verify, for the same level of noise, a slightly better performance in the first case. This is the expected behavior because, in the latter experiment, we had one or two outliers. Nevertheless, when the variance of noise is below 3 pixels, the projected error achieved by our approach was very similar to the *reference*.

As the synthetic experiments showed, the theoretical principle presented in this thesis can be

applied in noisy conditions. Considering acceptable noisy levels ($\sigma \leq 3$), the performance of our implementation is identical to the *reference*, mainly when we compare the projection error.

4.4.2 Real data

In order to evaluate the method’s behavior with real data, several data sets were used. In the first experiment, we used the Hotel sequence³. The 3D object has 106 points and the sequence is composed of 182 images (Figure 4.7a).

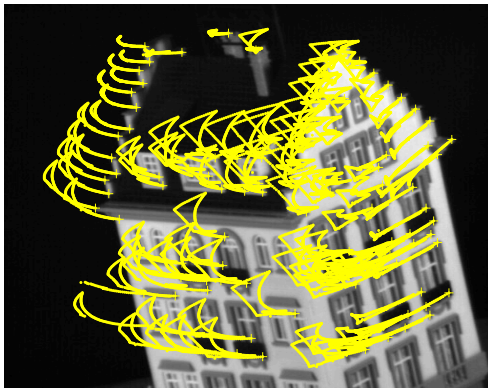
Using this data, 200 experiments were performed randomly selecting 5 (out of 182) images in each experiment. The object shape \mathbf{S} is computed from 4 images using Tomasi-Kanade shape-from-motion algorithm, and the other one is used as a test image $\overline{\mathbf{W}}$. By observing Figure 4.7b, we can see that the algorithm obtained the correct solution almost 35% of the time.

Although the percentage of “completely” correct solutions is not high (Figure 4.7b), a wrong solution does not imply a gross mistake, as it was underlined in the synthetic case. Evaluating the histograms of Figures 4.7b and 4.7c, we can check that the mean projection error is below 6 pixels when the algorithm output can have 16 wrong matches. Comparing the first part of histogram, we clearly see the fact that several wrong matches can not be always connected with gross mistakes. When the wrong matches of the obtained solution are less than or equal 6, the mean projection error is less than 1 pixel. As shown in Figure 4.8, common mistakes occur because neighboring points are switched.

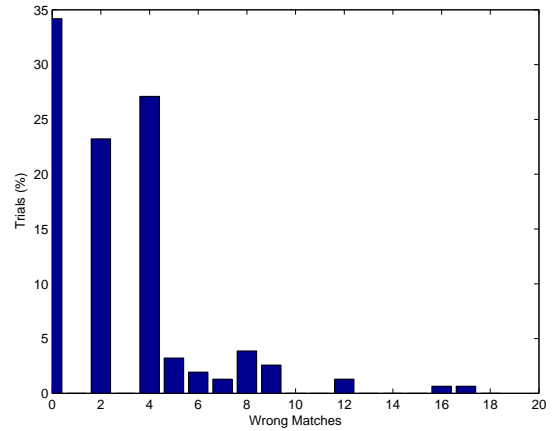
Such as Figure 4.7d clearly shows, the error function value of the expected and estimated solutions are almost the same however they are numerically different in more than 65% of the time. This happens because sometimes the images selected to compute the shape are almost degenerate (the frames are very close in time). In others words, the Hotel’s model, a 3D subspace, is not the most adequate one because the model computed from 4 images can be well described by a 2D subspace (a degenerate case).

The following experiments allow us to verify the performance our methodology in the presence

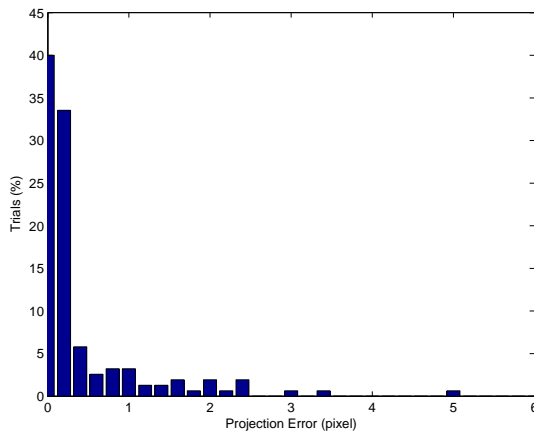
³<http://vasc.ri.cmu.edu/idb/html/motion/long-hotel/index.html>



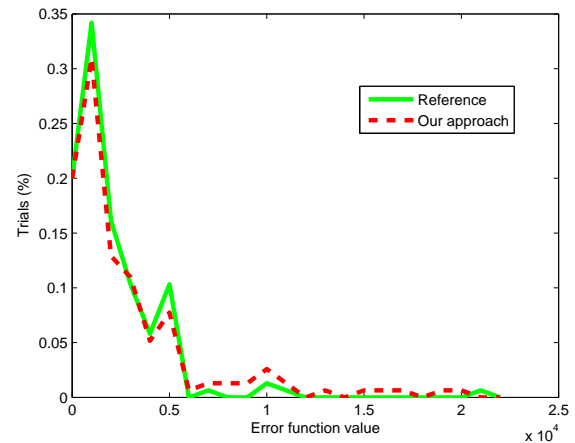
(a) Sequence with image tracks



(b) Histogram of wrong matches



(c) Histogram of mean projection error



(d) Distribution of the error function

Figure 4.7: Performance of the matching algorithm with real data: (a) shows the data set. We used the hotel sequence to generate the shape matrix. Images were randomly selected and matched against the 3D data. (b) Shows the histogram of the % of wrong matches for the whole range of experiments. Notice the very low frequency of high errors (17 in 106 points). However the impact of wrong matches is quite small as shown in (c): More than 80% of the trials had less than one pixel average error. (d) Even with errors, the error difference between the estimated permutation and the correct permutation is negligible.

of outliers and clutter points: one 2D-2D (image-to-image matching) case and two 2D-3D (image-to-model) matching will be shown. In all these cases, a repetitive pattern with salient corners (a grid) is used due to two main reasons:

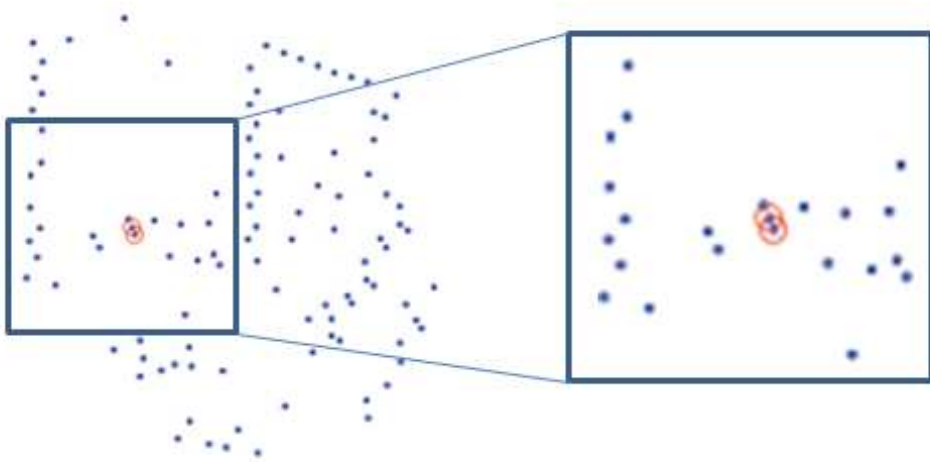


Figure 4.8: Common error: This figure illustrates the most common error. The two features are very close in the model and in the image. Quite often swapping decreases the value of the error function. Though accounted as an error it has no impact on real applications.

- the photometric-based algorithms' performance are poor because a large set of features are indistinguishable between them
- the features (corners) are easily detected by the standard corners detectors

The N feature coordinates shown in Figure 4.9 define a 2D object model - a N -dimensional rank 3 subspace. Observing the pictures of Figure 4.10, the smooth convergence of the algorithm can be checked, noticing that all points go fast and monotonically to the desired location. Note that points are initialized in the center of the image (green cloud). This is a surprising behavior since originally we had a 0-1 assignment problem. Here we are observing the evolution in a continuous domain and each iteration provides an estimate of point coordinates instead of a matching decision (only viable when the global optimum is reached - a permutation matrix). This fact explains the enormous gain of introducing the model-checking step (step 3 in Algorithm 4.3).

The next experiment demonstrates the whole potential of the methodology in tackling a problem of pose and correspondence determination. In more realistic situations data has outliers, sometimes with very large deviations. In the left picture of Figure 4.11, there are 4 data points in this situation.

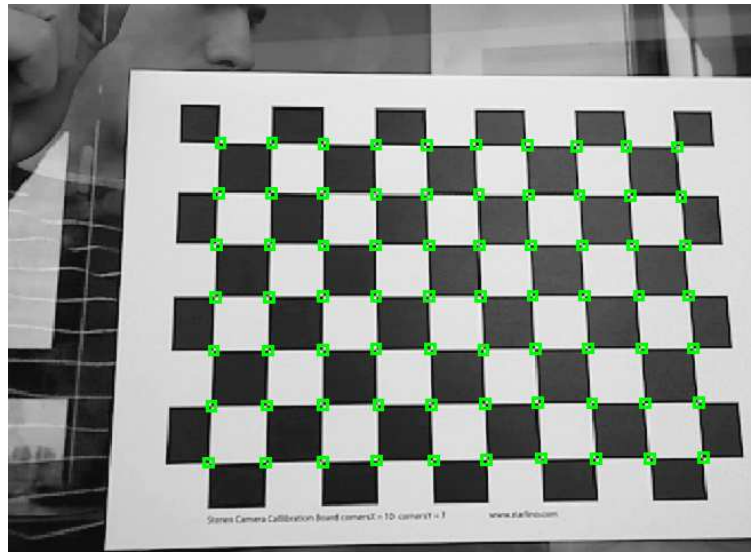


Figure 4.9: Image-to-Image matching: The green points show the 2D model. Points in other images are matched against this set.

The green squares represent data and red dots the estimated point locations ($\overline{\mathbf{WP}}$). Note also that since it is a N-to-N matching, every data point must have a corresponding model point. In this case, the algorithm was always able to provide a quite accurate estimate for the location of the outliers. Note the detail in the right picture of Figure 4.11 where the estimated points are very close to the correct location even if some data points (green squares) are not close to the corners. This is quite interesting because the algorithm is not affected by a few gross mistakes despite its global nature. Put in another way, there is empirical evidence that the algorithm has a much better convergence behavior for a subset of points, notably the most accurate ones. In this situation the minimizer is not a permutation since the outliers do not follow the model. However, the final \mathbf{P} has almost 1 in the correct points (order of 0.97) and for the outliers the convex combination provides an accurate position.

This robustness is exhibited in an even more difficult case, where we added 20% more clutter points. In other words, besides the correct points in the image we randomly spread 20% more points (15 in a total of 70 inliers).

Figure 4.12a shows the spatial distribution of the data points: green squares are inliers, red

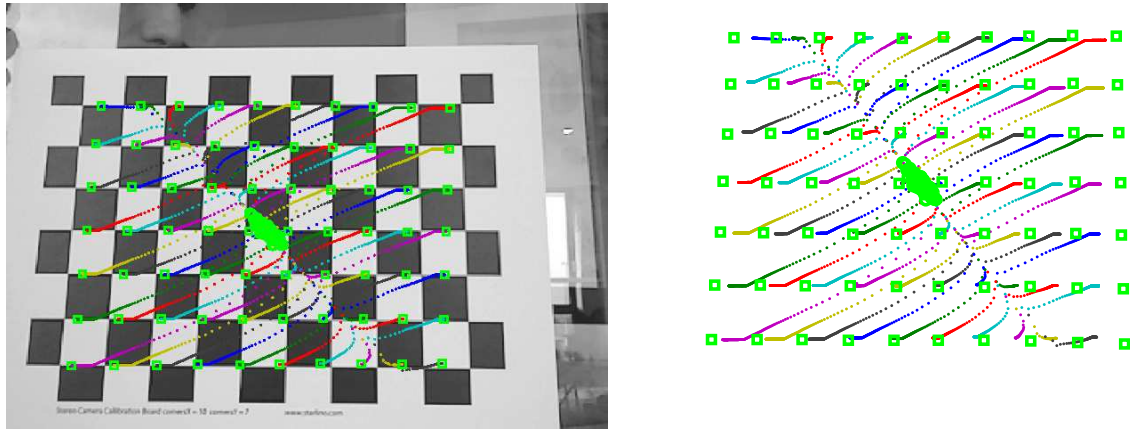


Figure 4.10: The trajectories show the convergence of the algorithm. Points were initialized in the central green cloud. After a few iterations their estimated coordinates exhibit a smooth and fast convergence in the direction of the correct position.

are clutter points and blue dots are the estimated feature positions $\overline{\mathbf{WP}}$. We initially sorted the points such that the initial 70×70 submatrix should be an identity matrix. Due to the noise and clutter the identity matrix is not exact but, as Figure 4.12b shows, it is very approximate. Most importantly, the \mathbf{P} matrix is block diagonal with $p_{ij} = 0$ if point i and j are image/model incompatible. The “redish” values are closer to 1 and the “blueish” closer to 0. Note that the lower-right 15×15 submatrix is not integer at all. These results enforce the usefulness of the GUI SAC strategy since it is noticeable that there are enough points with a correct assignment, thus allowing the model-checking step which solves the problem in one step.

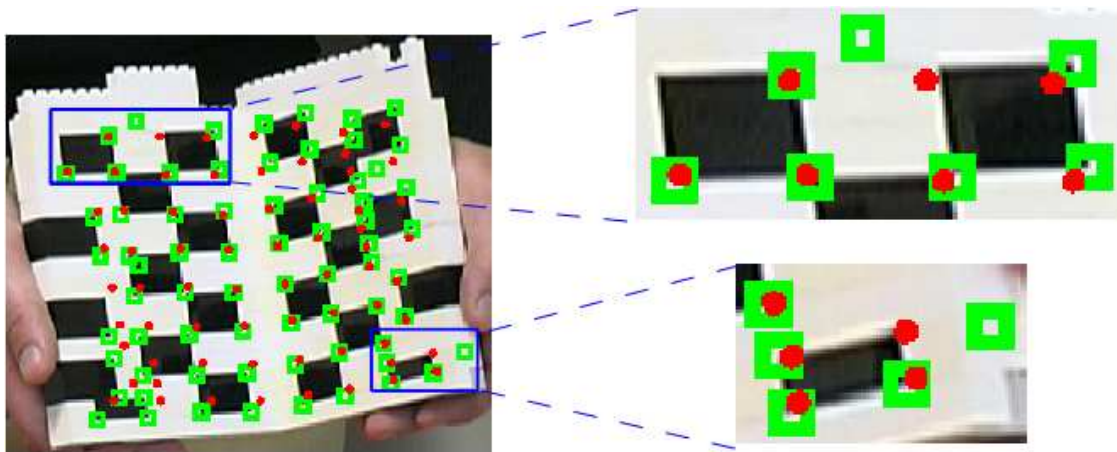
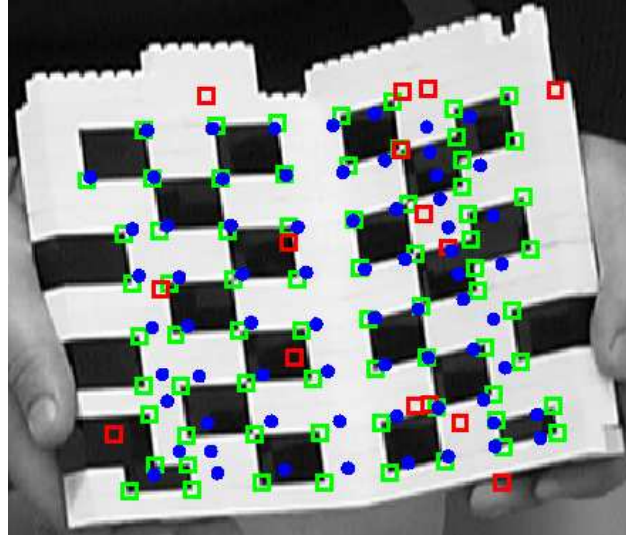
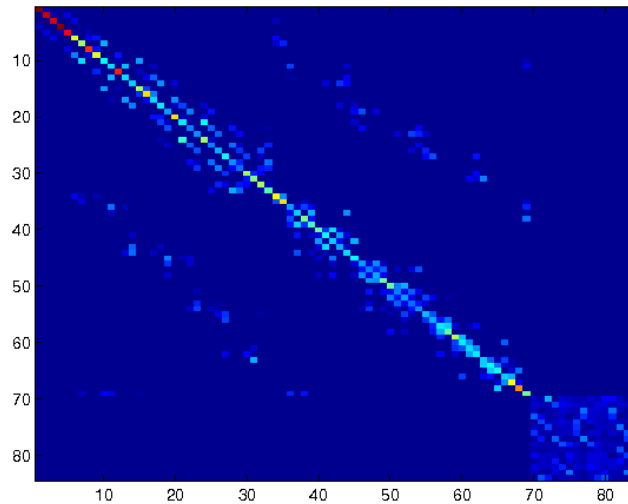


Figure 4.11: 2D-3D matching with outliers: Left - green squares represent data and red dots the estimated point locations ($\overline{\mathbf{WP}}$). Right - A detail of 2 outliers. The green squares should be on the corners. However the estimated point is very close to the correct location.



(a) Image data - green and red squares. Blue dots are estimated point coordinates



(b) The resulting \mathbf{P} (permutation) matrix after a few iterations

Figure 4.12: 2D-3D matching with clutter points: (a) - The squares are the data: green are inliers and red are clutter points. Blue dots are the estimated feature positions $\overline{\mathbf{WP}}$. (b) - The “redish” values are closer to 1 and the “blueish” closer to 0. Notice the 2 blocks forming an identity-like submatrix corresponding to the assignment of inliers.

Chapter 5

Conclusions

This thesis is composed by two main parts, addressing the following key challenges in Computer Vision:

- Estimating the 3D shape from an image sequence with missing data
- Finding the correspondence between two point sets, assuming that images are generated by an affine camera.

In both problems, the proposed algorithms explore the same foundational principle: world rigidity.

In spite of the same roots and similar structure, we considered the two problems separately: in the first one, point correspondences are known and we seek the 3D shape of the scene. In the second part, we compute the point matching, assuming known shape.

We showed and proved that, if the data contains degenerate frames, the rank-based algorithms are not able estimate 3D shape. Instead of a unique solution, the rank constraint leads to an infinite number of minimum error solutions. In order to avoid this situation, we imposed the full scaled-orthographic model to the data, thus replacing the rank constraint by the *motion manifold*.

Besides allowing computing the 3D shape with missing and degenerate data, this new constraint makes possible to compute the 3D shape with only two images (without missing data),

instead of the three required by factorization-based approaches, following Tomasi-Kanade's framework [Tomasi 92].

In the second part of the thesis, we introduced a new methodology to solve correspondence problems. Assuming an affine camera model, we demonstrated the conditions in order to obtain a unique solution for correspondence between two sets with the same number of points - the *Subspace Matching Theorem*. Based on this theoretical principle, we designed convex programs to find the global solution for matching problems rather than going thru combinatorial search. These convex problems were drawn such that the most recent gradient algorithms could be used.

Even though the theorem does not guarantee a unique solution when clutter points exist, the results show a high degree of resilience to these perturbations. This resilience is further increased because our approach can handle multiple matching criteria (or even support). By merging global geometric constraints with other matching measures (feature similarity, maximum disparity, smoothness) our method can seamlessly deal with clutter and outlier points.

We also developed a new strategy to speed up the minimization process, named GUI SAC due to the resemblance with RANSAC procedure. Exploiting the intrinsic nature of the matching problem, this strategy consists of an automatic selection of a small set of correspondences, anticipating the estimation of the affine transformation. Due to the initial fast convergence, this early estimation of the global mapping allows accurate assignments for all points in a fraction of the time, making this method suitable for real time application.

5.1 Discussion and Future Work

Since all developments were done in the context of affine/orthographic cameras, a straightforward upgrade is to extend our theoretical results and algorithms to incorporate the perspective camera model. However, from our point of view, this would not have a major impact. From the content of this thesis, we showed that our algorithms are able to provide acceptable initial points at a small computational cost, which can be further refined by perspective methods.

More significantly, in the context of the matching problem, if our approach could deal with

occlusion, outliers and clutter points, the scope of applicability increases exponentially. In the former two issues there is a correspondence problem with partial data and in the latter, a decision problem between inliers and outliers arises.

The most challenging issue to our formulation seems to be the inclusion of occluded and outlier points. This is due to our theory requiring all model points have one corresponding data point. Such a requirement is tightly coupled to the global signature of the model (a projection onto a linear subspace), thus, changing it may modify the deep structure of the problem. In our view, considering partial data as valid input would require a major restructuring of the theory and even its objectives: the definition of matching is ambiguous since defining a unique solution depends on how we treat the “non-matched” points. For example, if the number of requested matches is not specified, further regularization is needed to prevent trivial null solutions.

Within the scope of our theory, we foresee a clear extension of our formulation to cope with clutter points. As we showed in Chapter 4, selecting and matching inliers from cluttered data is reliably done through common convex optimization tools. The experimental results show that our framework is able to deal with this type of artifacts. One relevant direction of future research would be the embedding of this non-modeled data in the theoretical core. Because the uniqueness of the solution is not guaranteed, one possible way to tackle this issue is to discover new geometric constraints without reshaping the formulation from its roots. Appending these new geometric constraints to the optimization problem, the uniqueness of the solution may be guaranteed, avoiding the hard decision of discarding points. Alternatively, this question would be answered if we could prove that as experimental data suggests, the resulting minimizer is always block diagonal where each block, rather than matching points, segments inliers from outliers uniquely.

Another important topic to develop in the future is the design of optimization algorithms for matching problems in a convex framework. Based on the fact that some points converge much faster than others, speed up can be achieved by intrinsically coding this knowledge rather than using a model-checking approach (like GUI SAC).

Finally, and returning to Chapter 1, closing the loop between the reconstruction and matching

processes would be *quintessential* in order to solve the *3D Vision problem*. Figure 1.1 and expression 1.4 shed some light into it: reconstruct the shape with unknown correspondences or, in other words, match the data to the model without knowing the latter.

Appendix A

The *motion manifold* projection

A.1 Rigid factorization

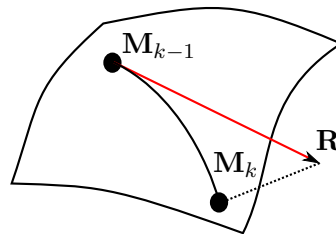


Figure A.1: Point \mathbf{R} is projected on the *motion manifold*

Algorithm 3 is an iterative algorithm which projects a matrix into the *motion manifold*, as Figure A.1 suggests.

This algorithm is a version of the power method [Hartley 03a] where in each iteration a *motion matrix* is calculated (step 2) independently. The procedure of step 2 projects the left factor of \mathbf{W}'_c onto the manifold of motion matrices (matrices with pairwise orthogonal rows). This projection has a similar derivation to the Procrustes problem [Golub] (see A.2) Specifically, we seek the matrix

Algorithm 3 Rigid Factorization

-
1. Initializations:
 (factorize \mathbf{W}'_c using any factorization (e.g. SVD))
 $\mathbf{W}'_c = \mathbf{A}\mathbf{B}$, $\mathbf{R} = \mathbf{A}$, $\mathbf{M}_0 = \mathbf{A}$, $\mathbf{S}_0 = \mathbf{B}$
 $k = 1$
 2. Project \mathbf{R} into the manifold of motion matrices
 $\mathbf{M}_k = \arg \min_{\mathbf{X}} \sum_f \|\mathbf{R}^f - \mathbf{X}^f\|_F^2$
 s. t. $\mathbf{X}^f \mathbf{X}^{fT} = \alpha^f \mathbf{I} \quad \forall f$
 $\alpha \in \mathbb{R}^+$
 3. $\mathbf{S}_k = \mathbf{M}_k^+ \mathbf{W}'_c$, \mathbf{M}_k^+ - Moore-Penrose pseudoinverse
 4. $\mathbf{R} = \mathbf{W}'_c \mathbf{S}_k^+$
 5. Verify if $\|\mathbf{M}_k - \mathbf{M}_{k-1}\| < \epsilon$.
 If not, go to step 2 and $k = k + 1$.
 6. $\mathbf{M} = \mathbf{M}_k$ and $\mathbf{S} = \mathbf{S}_k$
-

with pairwise orthogonal rows (not orthonormal) which is closest to the left factor of \mathbf{W}'_c , that is:

$$\begin{aligned}
 \mathbf{M}_k &= \arg \min_{\mathbf{X}} \|\mathbf{R} - \mathbf{X}\|_F^2 = \left\| \begin{bmatrix} R^1 \\ \vdots \\ R^F \end{bmatrix} - \begin{bmatrix} X^1 \\ \vdots \\ X^F \end{bmatrix} \right\|_F^2 \\
 &= \arg \min_{\mathbf{X}} \sum_f \|\mathbf{R}^f - \mathbf{X}^f\|_F^2 \\
 &\text{s. t. } \mathbf{X}^f (\mathbf{X}^f)^T = \alpha^f \mathbf{I} \quad \forall f, \alpha \in \mathbb{R}^+ \\
 \mathbf{M}_k^f &= \alpha^f U^f (V^f)^T, \text{ where } R^f = U^f \begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix} (V^f)^T \\
 &\text{and } \alpha^f = (\sigma_1 + \sigma_2)/2
 \end{aligned} \tag{A.1}$$

Even though step 2 is solved F times in each iteration, the computational cost is irrelevant because, as (A.1) shows, it has a closed-form solution and is unique (details in A.2). In step 3, the estimate of the object's shape is given by a Least-Square solution.

It is important to refer that matrix \mathbf{R}_k , used in step 2 and calculated in step 4, is not an algorithm's output, but an auxiliary matrix. This is used to adapt matrix \mathbf{S}_k to \mathbf{M}_k .

In terms of complexity, we need to compute two pseudo-inverses and a set of eigen vectors in closed-form (in \mathbb{R}^3).

A.2 The projection into scaled-Stiefel matrices

Considering the optimization problem given by

$$\begin{aligned} \text{Problem 8} \quad (\mathbf{X}, \alpha)^* &= \arg \min_{\mathbf{X}, \alpha} \|\mathbf{A} - \alpha \mathbf{X}\|_F^2 \\ &\text{s.t.} \quad \mathbf{X}\mathbf{X}^T = \mathbf{I} \end{aligned}$$

Applying the Forbenius norm definition, we have

$$\begin{aligned} (\mathbf{X}, \alpha)^* &= \arg \min_{\mathbf{X}, \alpha} \text{trace}(\mathbf{A}\mathbf{A}^T - 2\alpha\mathbf{A}\mathbf{X}^T + \alpha^2\mathbf{X}\mathbf{X}^T) \\ &\text{s.t.} \quad \mathbf{X}\mathbf{X}^T = \mathbf{I} \end{aligned} \tag{A.2}$$

Since the first term of the cost function is a constant and using the constraint in the third term, Problem 8 is equivalent to

$$\begin{aligned} (\mathbf{X}, \alpha)^* &= \arg \min_{\mathbf{X}, \alpha} \text{trace}(-2\alpha\mathbf{A}\mathbf{X}^T + \alpha^2\mathbf{I}) \\ &\text{s.t.} \quad \mathbf{X}\mathbf{X}^T = \mathbf{I} \end{aligned} \tag{A.3}$$

Solving (A.3) in order to \mathbf{X} and recurring to the *Singular Value Decomposition* of \mathbf{A} and \mathbf{X} , $\mathbf{A} = \mathbf{U}_\mathbf{A}\Sigma_\mathbf{A}\mathbf{V}_\mathbf{A}^T$ and $\mathbf{X} = \mathbf{U}_\mathbf{X}\Sigma_\mathbf{X}\mathbf{V}_\mathbf{X}^T$, (A.3) implies

$$\begin{aligned} (\mathbf{U}_\mathbf{X}, \Sigma_\mathbf{X}, \mathbf{V}_\mathbf{X})^* &= \arg \max_{\mathbf{U}_\mathbf{X}, \Sigma_\mathbf{X}, \mathbf{V}_\mathbf{X}} \text{trace}(\alpha\mathbf{U}_\mathbf{A}\Sigma_\mathbf{A}\mathbf{V}_\mathbf{A}^T\mathbf{V}_\mathbf{X}\Sigma_\mathbf{X}\mathbf{U}_\mathbf{X}^T) \\ &\text{s.t.} \quad \mathbf{U}_\mathbf{X}\Sigma_\mathbf{X}\mathbf{V}_\mathbf{X}^T\mathbf{V}_\mathbf{X}\Sigma_\mathbf{X}\mathbf{U}_\mathbf{X}^T = \mathbf{I} \end{aligned} \tag{A.4}$$

Using some proprieties of the trace and the Von Neumann's trace inequality [Mirsky 75], the maximum value of (A.4) is attained when $\mathbf{U}_\mathbf{X} = \mathbf{U}_\mathbf{A}$, $\mathbf{V}_\mathbf{X} = \mathbf{V}_\mathbf{A}$ and $\Sigma_\mathbf{X} = \mathbf{I}$. Given the solution $\mathbf{X}^* = \mathbf{U}_\mathbf{X}\mathbf{V}_\mathbf{X}^T$, the correspondent value of the cost function (A.4) is $2\alpha\Sigma_\mathbf{A}$.

Returning to (A.3), we have

$$\alpha^* = \arg \min_{\alpha} \text{trace}(-4\alpha\Sigma_{\mathbf{A}} + \alpha^2\mathbf{I}) \quad (\text{A.5})$$

Since $\text{trace}(\Sigma_{\mathbf{A}}) = \lambda_1 + \lambda_2$, α^* is given by

$$\alpha^* = \frac{\lambda_1 + \lambda_2}{2} \quad (\text{A.6})$$

In summary, the global optimum of Problem 8 can be written as

$$\mathbf{X}^* = \mathbf{U}_{\mathbf{A}}\mathbf{V}_{\mathbf{A}}^T \quad (\text{A.7})$$

$$\alpha^* = \frac{\lambda_1 + \lambda_2}{2} \quad (\text{A.8})$$

Appendix B

A 3D reconstruction from two images

Before we present our method to recover 3D shape from only two images, we will describe the common framework for sequences. The well-succeeded Tomasi-Kanade factorization method [Tomasi 92] allows us to estimate the 3D shape from an image stream under the orthographic camera model. If a factorization of the *data matrix* \mathbf{W} is available (e.g. *Singular Value Decomposition*), we have

$$\mathbf{W} = \mathbf{A}\mathbf{B} \tag{B.1}$$

where \mathbf{A} and \mathbf{B} are rank 3 matrices ($\mathbf{A}\mathbf{B} \in \mathcal{R}_3$). In order to compute the 3D shape \mathbf{S} , the factorization [Tomasi 92] imposes the camera model to the data through a global map \mathbf{Q} such that each row pair of $\mathbf{A}\mathbf{Q}$ complies with the orthographic model. This map is the solution of the following optimization problem

$$(\mathbf{Q})^* = \arg \min_{\mathbf{Q}} \sum_{f=1}^F \left\| \mathbf{A}^f \mathbf{Q} \mathbf{Q}^{-1} (\mathbf{A}^f)^T - \alpha^f \mathbf{I} \right\|_2^2 + \|\alpha^1 - 1\|_2^2 \tag{B.2}$$

where \mathbf{A}^f is composed by the rows $2f - 1$ and $2f$ of \mathbf{A} and α^f is the scale factor of image f . Note that the sequence has F images.

Considering $\mathbf{G} = \mathbf{Q}\mathbf{Q}^{-1}$, we obtain a Least-Squares solution given by

$$\begin{bmatrix} \mathbf{g} \\ \alpha^2 \\ \vdots \\ \alpha^F \end{bmatrix} = (\mathbf{K}\mathbf{K}^T)^{-1} \mathbf{K}\mathbf{b} \quad (\text{B.3})$$

where \mathbf{g} is $\text{vec}(\mathbf{G})$ and \mathbf{b} is (almost) a zero vector, except in the first and fourth positions. In these positions, it is equal to 1. Matrix \mathbf{K} can be divided in two parts as the equation shows.

$$\mathbf{K} = \begin{bmatrix} \mathbf{F} & \alpha_{\mathbf{v}}^2 & \dots & \alpha_{\mathbf{v}}^F \end{bmatrix}$$

Vectors $\alpha_{\mathbf{v}}^f$ are also composed by zeros except in the cells $2f + 1$ and $2f + 4$ where it is -1. The first part of \mathbf{K} , matrix \mathbf{F} , is given by

$$\mathbf{F} = \begin{bmatrix} \mathbf{A}^1 \otimes \mathbf{A}^1 \\ \vdots \\ \mathbf{A}^F \otimes \mathbf{A}^F \end{bmatrix} \quad (\text{B.4})$$

Each matrix \mathbf{A}^f is equal to the rows $2f - 1$ and $2f$ of \mathbf{A} . If only two images are available this methodology does not work because we have less constraints than variables to estimate: 8 constraints, each image generates 4 constraints (B.4), and 6 variables (\mathbf{G} is a 3×3 symmetric matrix). Since $(\mathbf{K}\mathbf{K}^T)$ does not have an inverse matrix, the solution expressed by (B.3) is not computable.

The non-convex optimization problem is efficiently solved by the algorithm described in Appendix A.1. To verify the performance of this method, we estimated the 3D Hotel shape from two images of the dataset (Figure B.1).

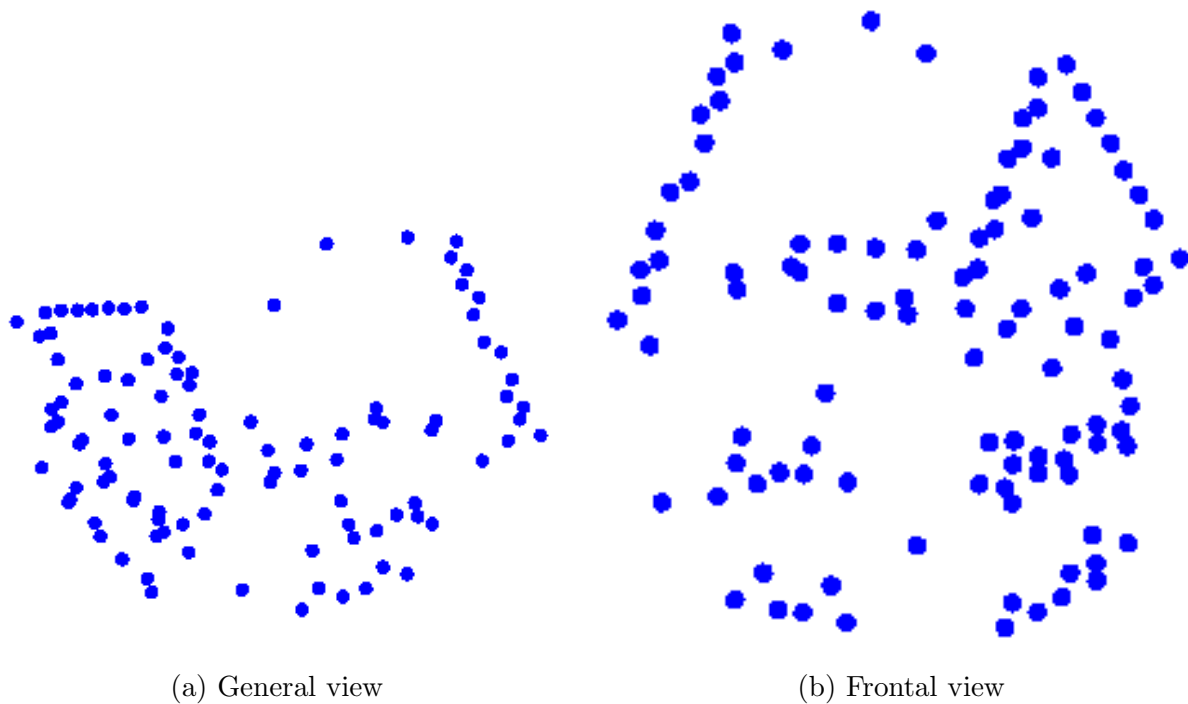


Figure B.2: A 3D reconstruction from two images: different views of the estimated shape

Checking the pictures of Figure B.2, we can verify that the proposed allows us to obtain a reasonable shape reconstruction. As expected, the reconstruction performed from two images is more dependent of slight pixel deviations than a reconstruction from the whole sequence. This can be verified in Figure B.3 where the reconstruction from two images (blue squares) exhibits a considerable error comparing with the ground-truth (black dots or red circles).

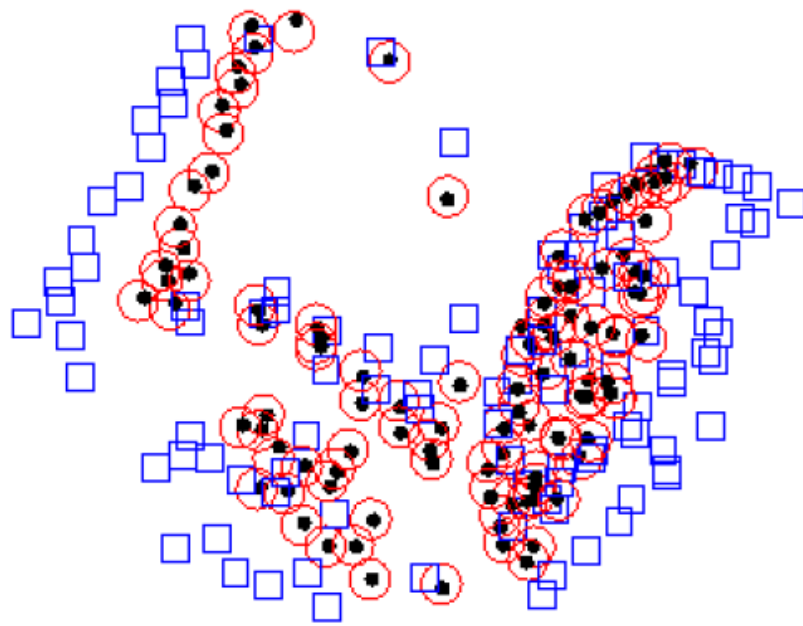


Figure B.3: A 3D reconstruction from two images: Comparing with reconstruction performed from the whole image sequence. Black dots - Tomasi-Kanade from the whole sequence. Red dots - Rigid Factorization from the whole sequence. Blue squares - Rigid Factorization from two images. The reconstructions computed by Tomasi-Kanade and Rigid Factorization from the whole sequence can be considered as ground-truth

Appendix C

The orthographic solutions

Considering the *shape matrix* known and rank constraints, we have the following system of equations

$$\underbrace{\begin{bmatrix} (\mathbf{W}'_c)_{[2f-1,1]} & \cdots & (\mathbf{W}'_c)_{[2f-1,m]} & ? & \cdots & ? \\ (\mathbf{W}'_c)_{[2f,1]} & \cdots & (\mathbf{W}'_c)_{[2f,m]} & ? & \cdots & ? \end{bmatrix}}_{\mathbf{W}'_c} = \underbrace{\begin{bmatrix} a & b & c \\ d & e & f \end{bmatrix}}_{\mathbf{M}^f} \mathbf{A} \quad (\text{C.1})$$

$$\mathbf{A}^{-1} \underbrace{\begin{bmatrix} \mathbf{S}_{11} & \cdots & \mathbf{S}_{1m} & \mathbf{S}_{[1,m+1]} & \cdots & \mathbf{S}_{1N} \\ \mathbf{S}_{21} & \cdots & \mathbf{S}_{2m} & \mathbf{S}_{[2,m+1]} & \cdots & \mathbf{S}_{2N} \\ 0 & \cdots & 0 & \mathbf{S}_{[3,m+1]} & \cdots & \mathbf{S}_{3N} \end{bmatrix}}_{\mathbf{S}}$$

From (C.1) the variables a , b , c and d are calculated, instead of e and f which have infinite solutions. To make the solutions space smaller, the orthogonality constraints, given by the following equations, are added.

$$\begin{cases} a^2 + b^2 + c^2 = \alpha^2 \\ d^2 + e^2 + f^2 = \alpha^2 \\ ad + be + cf = 0 \end{cases} \quad (\text{C.2})$$

Note that only e and f are variables because a , b , c and d are determined by (C.1). Considering only the first two equations of (C.2), we can write

$$\begin{cases} a^2 + b^2 + c^2 &= d^2 + e^2 + f^2 \\ ad + be + cf &= 0 \end{cases} \quad (\text{C.3})$$

In order to obtain c and f , the last equation is solved for both variables,

$$\begin{cases} c'^2 + (a^2 + b^2 - d^2 - e^2) c' - a^2 d^2 - b^2 e^2 - 2adbe = 0 \\ f'^2 + (-a^2 - b^2 + d^2 + e^2) f' - a^2 d^2 - b^2 e^2 - 2adbe = 0 \end{cases} \quad (\text{C.4})$$

To simplify the notation, it is important to note that these 2nd order equations (C.4) were obtained by the following change of variables: $c' = c^2$ and $f' = f^2$. Due to the similarity between the equations, the following steps aim to calculate c , because f is obtained in the same way. Then, the roots of first equation are

$$c' = \frac{1}{2} \left(-a^2 - b^2 + d^2 + e^2 \pm \sqrt{gh} \right)$$

where $g = d^2 + 2db + b^2 + e^2 - 2ea + a^2$ and $h = d^2 - 2db + b^2 + e^2 + 2ea + a^2$. Because of $c' = c^2$ and $c \in \mathbb{R}$, c' is, necessarily, 0 or a positive value. Due to this we consider the first root. Then, the value of c is given by

$$c = \pm \sqrt{\frac{1}{2} \left(-a^2 - b^2 + d^2 + e^2 + \sqrt{gh} \right)} \quad (\text{C.5})$$

Through an analogous way, f is given by

$$f = \mp \sqrt{\frac{1}{2} \left(a^2 + b^2 - d^2 - e^2 + \sqrt{gh} \right)} \quad (\text{C.6})$$

Instead of an infinite number of solutions allowed by the affine camera model, only two solutions are according to orthogonality constraints.

As Figure C.1, these two solutions correspond to the correct one, and the reflection of the camera over the plane (shape). This reflection produces the same image (it is intrinsic to orthography). Even though motion can be “reflected”, the computed shape will always be correct, up to the ambiguity of an orthogonal transformation, inherent to the orthographic model. This means both that the Motion and Shape matrix fit the known data and also that the Motion matrix satisfies the orthogonality constraints. Note that, if the known data is over a line (the known part of shape is rank 1), there will be infinite solutions for motion. However, this will not affect the shape either (orthography is valid).

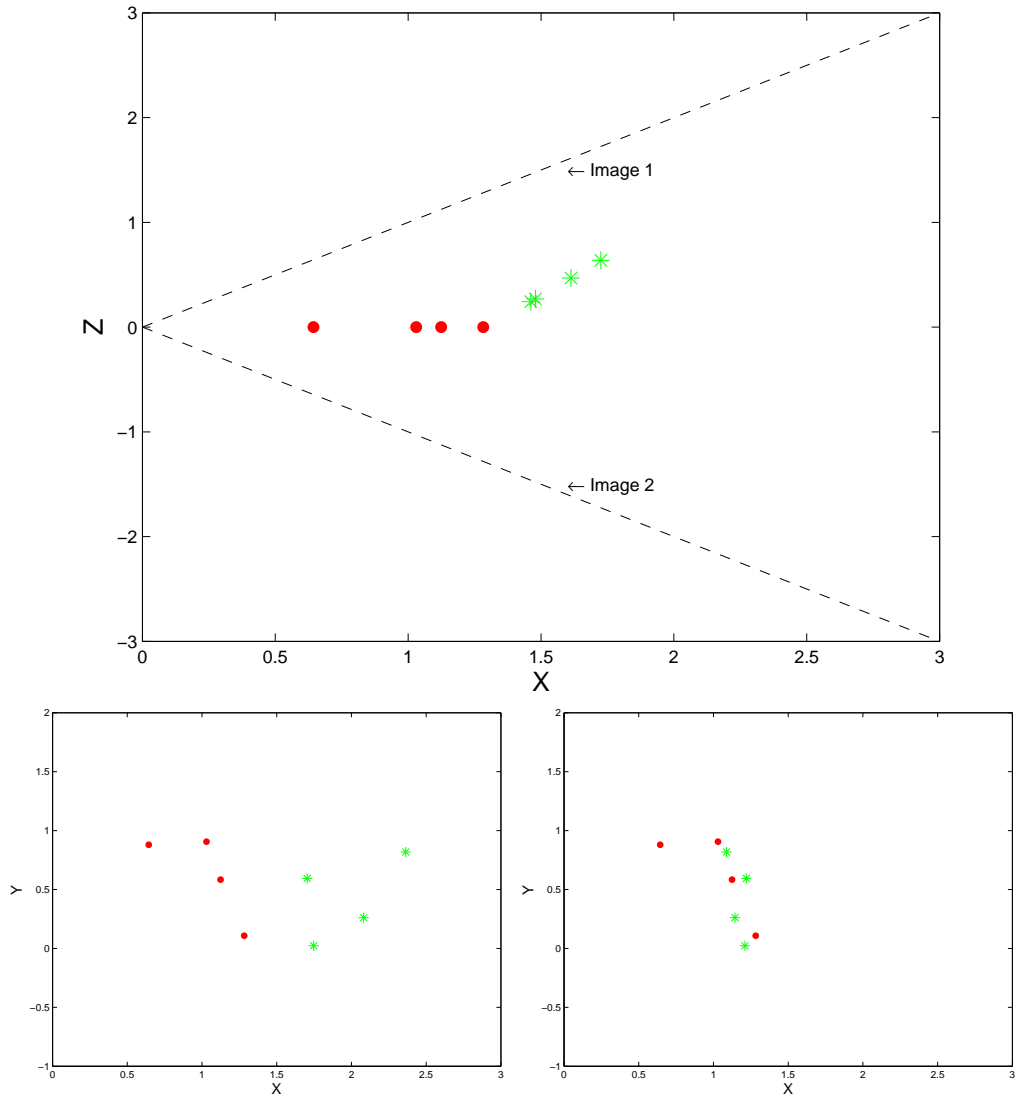


Figure C.1: Two solutions: the known points in frame f are represented by red circles and the unknown points by green circles. Bottom Left - 2D projections in image 1. Bottom Right - 2D projections in image 2.

Appendix D

The subspace matching theorem

Let $x_1, \dots, x_N \in R^k$. And let

$$D = \begin{bmatrix} x_1 & \cdots & x_N \end{bmatrix}^T \in R^{N \times k}. \quad (\text{D.1})$$

Let S be a subspace of R^N , with $\text{rank } S = r$, where $r \geq k$.

Consider the following problem:

Problem 1 *Suppose that there exists a permutation matrix Π such that the columns of ΠD belong to S . Find the matrix Π .*

Also, it is related to the classical correspondence problem of finding a permutation that transforms one collection of points in R^k into another one (see Section D).

For large N (which is a standard case in the applications), this problem is very hard to resolve. Because of that, we shall consider its important particular case. In fact, from now on we shall

consider permutation matrix Π to be of the following form

$$\Pi := \begin{bmatrix} I_{r-k+1} & 0 \\ 0 & \Pi' \end{bmatrix}, \quad (\text{D.2})$$

where now $\Pi' \in R^{(N-r+k-1) \times (N-r+k-1)}$ is a permutation matrix.

We can now formulate our main problem:

Problem 2 *Suppose that there exists a permutation matrix Π of the form (D.2), such that the columns of ΠD belong to S . Find the matrix Π .*

Moreover, in Section D, we shall discuss how the solution of Problem 2 gives an algorithm for a solution of the general case (Problem 1).

Without loss of generality, we are going to study these problems in a generic case. In particular, we have certain generic conditions on matrix D and on subspace S (precise conditions are given in Section D).

Our approach to Problem 2 is to study doubly-stochastic matrices, instead of permutation matrices.

Let

$$M' := \begin{bmatrix} I_{r-k+1} & 0 \\ 0 & M \end{bmatrix}, \quad (\text{D.3})$$

where $M \in R^{(N-r+k-1) \times (N-r+k-1)}$ is a doubly-stochastic matrix. Now, the analogous problem to Problem 2, for doubly-stochastic matrices is the following one:

Problem 3 *Suppose that there exists a doubly stochastic matrix M' of the form (D.3), such that the columns of $M' D$ belong to S . Find the matrix M' .*

The last problem can be resolved quickly and efficiently, see e.g. [Marques 09]. So the goal of this paper is to use the solution of Problem 3 to resolve Problem 2. In fact, we prove the following:

Let Π be a permutation matrix of the form (D.2), such that the columns of ΠD belong to S .

If M' is a doubly-stochastic matrix of the form (D.3), such that the columns of $M' D$ belong to S , then $M' = \Pi$.

This is the main result of the paper given in Theorem 2 from Section D. The proof relies on the Perron-Frobenius theorem applied for the special case of doubly-stochastic matrices. This theorem gives the uniqueness of the permutation matrix Π with the wanted properties in the set of all doubly-stochastic matrices.

Notation and auxiliary results

In this section we recall some of the properties of doubly-stochastic matrices that will be essential for the rest of the paper. These are classical results that are based on the theory of non-negative matrices and the Perron-Frobenius theorem [Minc 88], applied to the particular case of doubly-stochastic matrices.

Let M be a doubly-stochastic matrix, i.e. a non-negative square matrix whose row and column sums are all equal to 1. There exists a permutation matrix P such that PMP^T has the following lower block-triangular form:

$$PMP^T = \begin{bmatrix} M_1 & 0 & \cdots & 0 \\ M_{21} & M_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ M_{l1} & M_{l2} & \cdots & M_l \end{bmatrix},$$

where M_1, \dots, M_l , are square irreducible matrices. Moreover, since M is a non-negative matrix, with row and column sums equal to 1, we have that all off-diagonal blocks are equal to zero, i.e. all

block M_{ij} with $i > j$ are zero matrices. So we have that for the doubly-stochastic matrix M , the matrix PMP^T has the block-diagonal form:

$$PMP^T = \begin{bmatrix} M_1 & 0 & \cdots & 0 \\ 0 & M_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & M_l \end{bmatrix} \quad (\text{D.4})$$

with all blocks being irreducible.

The following lemma is classical and it is an application of the Perron-Frobenius theorem for irreducible doubly-stochastic matrices. It will be essential in the proof of the main result.

Lemma 2 *If M is an irreducible doubly-stochastic matrix, and w is a vector such that $Mw = w$, then all entries of w are equal, i.e. there exists $c \in \mathbb{R}$, such that*

$$w = c \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}.$$

Generic conditions

Let N , r and k be positive integers, such that $r \geq k$, and $N \geq r + 2(k - 1)$. Let

$$D = \begin{bmatrix} a^1 & a^2 & \cdots & a^k \\ X^1 & X^2 & \cdots & X^k \end{bmatrix} \in \mathbb{R}^{N \times k}, \quad (\text{D.5})$$

be a matrix where $a^i \in \mathbb{R}^{(r-k+1) \times 1}$ and $X^i \in \mathbb{R}^{(N-r+k-1) \times 1}$, $i = 1, \dots, k$ are given vectors. We also denote

$$X = \begin{bmatrix} X^1 & X^2 & \cdots & X^k \end{bmatrix} \in \mathbb{R}^{(N-r+k-1) \times k}. \quad (\text{D.6})$$

Let S be a prescribed r -dimensional vector subspace of \mathbb{R}^N , such that the columns of D belong to S . Then S is the span of some r linearly independent vectors:

$$S = \left\langle \begin{bmatrix} e^1 \\ f^1 \end{bmatrix}, \begin{bmatrix} e^2 \\ f^2 \end{bmatrix}, \dots, \begin{bmatrix} e^r \\ f^r \end{bmatrix} \right\rangle, \quad (\text{D.7})$$

where $e^i \in \mathbb{R}^{r-k+1}$, $f^i \in \mathbb{R}^{N-r+k-1}$, $i = 1, \dots, r$. We denote by B the corresponding matrix:

$$B = \begin{bmatrix} e^1 & e^2 & \dots & e^r \\ f^1 & f^2 & \dots & f^r \end{bmatrix} \in \mathbb{R}^{N \times r}. \quad (\text{D.8})$$

We shall assume that the matrix D and the subspace S , satisfy the following two conditions:

Condition 3 *The projection of S onto the $(r - k + 1)$ -dimensional subspace of \mathbb{R}^N , formed by the first $r - k + 1$ coordinates, is surjective. In other words, the matrix*

$$[I_{r-k+1} \ 0]B = [e^1, \dots, e^r],$$

is of full row-rank, i.e. has rank $r - k + 1$. By reordering, without loss of generality, we can assume that the vectors $e^1, e^2, \dots, e^{r-k+1}$ are linearly independent, i.e. that the matrix

$$Q := \begin{bmatrix} e^1 & e^2 & \dots & e^{r-k+1} \end{bmatrix},$$

is invertible.

Moreover, let v_1, \dots, v_{k-1} be vectors defined by:

$$v_i := f^{r-i+1} - \begin{bmatrix} f^1 & \dots & f^{r-k+1} \end{bmatrix} Q^{-1} e^{r-i+1}, \quad i = 1, \dots, k-1. \quad (\text{D.9})$$

We furthermore require that all submatrices of the matrix $[v_1 \ \dots \ v_{k-1}]$ formed by any $k - 1$ of its rows have rank $k - 1$.

Condition 4 Let T be the set of all $(N - r + k - 1) \times (N - r + k - 1)$ matrices R , such that all entries of R are 0's and 1's, with exactly one 1 in each row. We require that for all matrices $R \in T$ and every d such that $k \leq d \leq N - r + k - 1$, we have the following: for arbitrary $1 \leq i_1 < i_2 < \dots < i_d \leq N - r + k - 1$, the submatrix of $[X \ R]$ formed by the rows i_1, i_2, \dots, i_d has the rank $k + p$, where p is the number of the nonzero columns in the submatrix of R formed by the rows i_1, i_2, \dots, i_d , whenever $d \geq k + p$.

Since the set T has finitely many elements, it is straightforward to see that generic matrix D and subspace S satisfy the Conditions 3 and 4. In other words, the sets of matrices D and B that do not satisfy Conditions 4 and 3, respectively, are of measure zero.

Main result

Before proving the main result, we give a solution to the following theorem:

Theorem 5 Let $D \in \mathbb{R}^{N \times k}$ be a matrix (D.5) and let S be a r -dimensional subspace of \mathbb{R}^N (D.7) containing the columns of D , such that D and S satisfy the Conditions 3 and 4. Then we have the following:

If $M \in \mathbb{R}^{(N-r+k-1) \times (N-r+k-1)}$ is a doubly-stochastic matrix such that all vectors $\begin{bmatrix} a^i \\ MX^i \end{bmatrix}$, $i = 1, \dots, k$, belong to S , then M is the identity matrix.

Proof:

Our first and main goal is to find a vector w which is a nonzero linear combination of the vectors X^1, X^2, \dots, X^k such that

$$Mw = w. \tag{D.10}$$

Having this relation, by using Lemma 2 we shall obtain strong restrictions on w .

In order to find a vector w that satisfies (D.10) we do the following: denote by Σ^j , $j = 1, \dots, k$, the space of all vectors of the form $\begin{bmatrix} a^j \\ Z^j \end{bmatrix}$, where Z^j runs through $\mathbb{R}^{(N-r+1) \times 1}$, which belong to S , i.e.:

$$\Sigma^j = \left\{ \begin{bmatrix} a^j \\ Z^j \end{bmatrix} \mid Z^j \in \mathbb{R}^{N-r+k-1} \right\} \cap S.$$

Let $j \in \{1, \dots, k\}$ be fixed. We have that $\begin{bmatrix} a^j \\ Z^j \end{bmatrix} \in S$ if and only if there exists $\alpha_i^j \in \mathbb{R}$, $i = 1, \dots, r$, such that

$$\begin{bmatrix} a^j \\ Z^j \end{bmatrix} = \sum_{i=1}^r \alpha_i^j \begin{bmatrix} e^i \\ f^i \end{bmatrix},$$

i.e. such that

$$a^j = \sum_{i=1}^r \alpha_i^j e^i, \quad (\text{D.11})$$

$$Z^j = \sum_{i=1}^r \alpha_i^j f^i. \quad (\text{D.12})$$

Since S satisfies the Condition 3, the matrix

$$Q = [e^1 \ e^2 \ \dots \ e^{r-k+1}]$$

is invertible. Then (D.11) becomes

$$a^j = [e^1 \ e^2 \ \dots \ e^r] \begin{bmatrix} \alpha_1^j \\ \vdots \\ \alpha_r^j \end{bmatrix} = Q \begin{bmatrix} \alpha_1^j \\ \vdots \\ \alpha_{r-k+1}^j \end{bmatrix} + \sum_{l=r-k+2}^r \alpha_l^j e^l,$$

and so

$$\begin{bmatrix} \alpha_1^j \\ \vdots \\ \alpha_{r-k+1}^j \end{bmatrix} = Q^{-1}a^j - \sum_{l=r-k+2}^r \alpha_l^j Q^{-1}e^l. \quad (\text{D.13})$$

By replacing this in (D.12) we obtain that

$$\begin{aligned} Z^j &= [f^1 \ f^2 \ \dots \ f^r] \begin{bmatrix} \alpha_1^j \\ \vdots \\ \alpha_r^j \end{bmatrix} = [f^1 \ \dots \ f^{r-k+1}] \begin{bmatrix} \alpha_1^j \\ \vdots \\ \alpha_{r-k+1}^j \end{bmatrix} + \sum_{l=r-k+2}^r \alpha_l^j e^l = \\ &= [f^1 \ \dots \ f^{r-k+1}] Q^{-1}a^j + \sum_{l=r-k+2}^r \alpha_l^j (f^l - [f^1 \ \dots \ f^{r-k+1}] Q^{-1}e^l). \end{aligned}$$

In (D.9) we defined the vectors $v_1, \dots, v_{k-1} \in \mathbb{R}^{N-r+k-1}$ as

$$v_i = f^{r-i+1} - [f^1 \ \dots \ f^{r-k+1}] Q^{-1}e^{r-i+1}, \quad i = 1, \dots, k-1. \quad (\text{D.14})$$

Then we have that Z^j belongs to the affine subspace spanned by v_1, \dots, v_{k-1} . Finally, since

$$\begin{bmatrix} a^j \\ X^j \end{bmatrix} \in S, \text{ we have:}$$

$$\begin{bmatrix} a^j \\ Z^j \end{bmatrix} \in S \Leftrightarrow Z^j = X^j + \sum_{i=1}^{k-1} c_i^j v_i, \text{ for some } c_i^j \in \mathbb{R}. \quad (\text{D.15})$$

Note that the vectors v_i , $i = 1, \dots, k-1$ are independent on j , and hence all $\Sigma^1, \dots, \Sigma^k$ are parallel.

Now, let M be a doubly-stochastic matrix such that $\begin{bmatrix} a^j \\ MX^j \end{bmatrix} \in S$, for all $i = 1, \dots, k$. Then by (D.15) there exist real numbers $c_i^j \in \mathbb{R}$, $j = 1, \dots, k$, $i = 1, \dots, k-1$, such that

$$MX^j = X^j + \sum_{i=1}^{k-1} c_i^j v_i, \quad j = 1, \dots, k. \quad (\text{D.16})$$

Denote

$$V := \begin{bmatrix} v_1 & v_2 & \cdots & v_{k-1} \end{bmatrix} \in \mathbb{R}^{(N-r+k-1) \times (k-1)},$$

$$c^j := \begin{bmatrix} c_1^j \\ \vdots \\ c_{k-1}^j \end{bmatrix} \in \mathbb{R}^{(k-1) \times 1}, \quad j = 1, \dots, k,$$

$$C := \begin{bmatrix} c^1 & c^2 & \cdots & c^k \end{bmatrix} \in \mathbb{R}^{(k-1) \times k}.$$

Then (D.16) becomes

$$MX^j = X^j + Vc^j, \quad j = 1, \dots, k. \quad (\text{D.17})$$

Moreover, since the number of rows of the matrix C is $k-1$, we have that $\text{rank } C \leq k-1 < k$, and so the equation

$$C \begin{bmatrix} \beta_1 \\ \vdots \\ \beta_k \end{bmatrix} = 0,$$

has a nonzero solution $\begin{bmatrix} \beta_1 \\ \vdots \\ \beta_k \end{bmatrix}$. Thus these β_i 's satisfy:

$$\sum_{i=1}^k \beta_i c^i = 0. \quad (\text{D.18})$$

Now let

$$w := \sum_{i=1}^k \beta_i X^i.$$

Then from (D.17) and (D.18) we obtain

$$Mw = w + \sum_{j=1}^k \beta_j V c^j = w + V \left(\sum_{j=1}^k \beta_j c^j \right) = w. \quad (\text{D.19})$$

Thus, such defined w is an eigenvector of M , as wanted in (D.10).

Since M is a doubly-stochastic matrix, there exists a permutation matrix $P \in \mathbb{R}^{(N-r+k-1) \times (N-r+k-1)}$ such that

$$PMP^T = \begin{bmatrix} M_1 & 0 & \cdots & 0 \\ 0 & M_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & M_l \end{bmatrix} \quad (\text{D.20})$$

with all diagonal blocks being irreducible. Also, we denote the size of M_i by d_i , $i = 1, \dots, l$, and we assume that $d_1 \geq d_2 \geq \dots \geq d_l \geq 1$. If $d_1 = 1$ (and consequently $d_i = 1$, for all $i = 1, \dots, l$), then $M = I$, as wanted. Otherwise, let $p := \max\{i \mid d_i \geq 2\}$, with the convention that $d_0 = +\infty$. Our aim is to prove that $p = 0$, which would finish our proof.

To that end, define vectors $x^j := PX^j$, $j = 1, \dots, k$, and $\omega := Pw = \sum_{j=1}^k \beta_j x^j$, and split them accordingly with respect to (D.20), i.e.:

$$x^j = \begin{bmatrix} x_1^j \\ \vdots \\ x_l^j \end{bmatrix}, \quad j = 1, \dots, k, \quad \omega = \begin{bmatrix} \omega_1 \\ \vdots \\ \omega_l \end{bmatrix},$$

with x_i^j , $\omega_i \in \mathbb{R}^{d_i}$, $i = 1, \dots, l$. Then by (D.19) we have that for all $i = 1, \dots, l$:

$$M_i \omega_i = \omega_i.$$

By Lemma 2, we have that there exist $k_i \in \mathbb{R}$, $i = 1, \dots, p$, such that

$$\omega_i = k_i \mathbf{1}_{d_i}, \quad i = 1, \dots, p,$$

where $\mathbf{1}_{d_i} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \in \mathbb{R}^{d_i}$. Then

$$\sum_{j=1}^k \beta_j x_i^j = c_i \mathbf{1}_{d_i}, \quad i = 1, \dots, p.$$

If we denote

$$y^j := \begin{bmatrix} x_1^j \\ x_2^j \\ \vdots \\ x_p^j \end{bmatrix}, \quad j = 1, \dots, k,$$

we obtain that

$$\sum_{j=1}^k \beta_j y^j - k_1 \begin{bmatrix} \mathbf{1}_{d_1} \\ \mathbf{0}_{d_2} \\ \vdots \\ \mathbf{0}_{d_p} \end{bmatrix} - k_2 \begin{bmatrix} \mathbf{0}_{d_1} \\ \mathbf{1}_{d_2} \\ \vdots \\ \mathbf{0}_{d_p} \end{bmatrix} - \dots - k_p \begin{bmatrix} \mathbf{0}_{d_1} \\ \mathbf{0}_{d_2} \\ \vdots \\ \mathbf{1}_{d_p} \end{bmatrix} = 0,$$

i.e. since not all β_i 's are equal to zero, we have that $k+p$ vectors $y^1, y^2, \dots, y^k, \begin{bmatrix} \mathbf{1}_{d_1} \\ \mathbf{0}_{d_2} \\ \vdots \\ \mathbf{0}_{d_p} \end{bmatrix}, \begin{bmatrix} \mathbf{0}_{d_1} \\ \mathbf{1}_{d_2} \\ \vdots \\ \mathbf{0}_{d_p} \end{bmatrix}, \dots, \begin{bmatrix} \mathbf{0}_{d_1} \\ \mathbf{0}_{d_2} \\ \vdots \\ \mathbf{1}_{d_p} \end{bmatrix}$ are linearly dependent.

However, since the matrix X satisfies the Condition 4, we have that $\sum_{i=1}^p d_i < k + p$, i.e.

$$\sum_{i=1}^l (d_i - 1) < k. \quad (\text{D.21})$$

In particular we have that $p \leq k - 1$, and so $\sum_{i=1}^p d_i \leq 2(k - 1)$. Thus PMP^T is of the form:

$$PMP^T = \begin{bmatrix} M' & 0 \\ 0 & I \end{bmatrix}, \quad (\text{D.22})$$

where M' is $2(k - 1) \times 2(k - 1)$ matrix.

Now, if all X^j , $j = 1, \dots, k$ satisfy $MX^j = X^j$, then by Lemma 2, the matrix X would have at least two identical rows, which contradicts the Condition 4.

So, there exists some j , such that $MX^j \neq X^j$. Without loss of generality, we can assume that $j = 1$. Then from (D.16) we would have that

$$PMP^T x^1 - x^1 = \sum_{i=1}^{k-1} c_i^1 P v_i.$$

However by (D.22), we have that the left-hand side of the above equation is a nonzero vector (and so not all c_i^1 's are equal to zero), whose all entries except the first $2(k - 1)$ ones are equal to zero. If we denote by $\nu_i \in \mathbb{R}^{N-r-k+1}$, $i = 1, \dots, k - 1$, the vectors formed by the last $N - r + k - 1 - 2(k - 1) = N - r - k + 1$ components of the vector $P v_i$, respectively, then the last implies that the vectors ν_1, \dots, ν_{k-1} are linearly dependent. However, this contradicts the second part of Condition 3 since $N - r - k + 1 \geq k - 1$.

Thus $M = I$, as wanted. ■

Finally, we can give our main result:

Theorem 6 *Let $D \in \mathbb{R}^{N \times k}$ be a matrix (D.5) and let S be a r -dimensional subspace of \mathbb{R}^N (D.7) such that D and S satisfy the Conditions 3 and 4.*

Let Π be a permutation matrix of the form (D.2), such that the columns of ΠD belong to S .

If M' is a doubly-stochastic matrix of the form (D.3), such that the columns of $M' D$ belong to S , then $M' = \Pi$.

Proof:

Let $D' = \Pi D = \begin{bmatrix} I_{r-k+1} & 0 \\ 0 & \Pi' \end{bmatrix} D$. Then the columns of D' belong to S , and also the columns of

$$\begin{bmatrix} I_{r-k+1} & 0 \\ 0 & M\Pi'^{-1} \end{bmatrix} D' = \begin{bmatrix} I_{r-k+1} & 0 \\ 0 & M \end{bmatrix} D,$$

belong to S . Finally, since $M\Pi'^{-1}$ is also a doubly-stochastic matrix and since D' satisfies the Condition 4, by Theorem 5 we have that $M\Pi'^{-1} = I$, i.e. $M = \Pi'$, as wanted. ■

Applications of the main result

In all problems posed and resolved in this appendix, we assume the existence of a certain permutation matrix, and the problem is in finding it. Moreover, our solution also detects whether such a permutation matrix exists. Indeed, the solution of Problem 3 determines whether there exists a wanted doubly-stochastic matrix, and if so, computes it. So if it doesn't exist or if the obtained doubly-stochastic matrix is not a permutation matrix, we conclude that a permutation matrix with the wanted properties does not exist.

Solution of the Problem 1

One approach to Problem 1 would be a search through all $N!$ permutation matrices.

With the approach given in this paper instead of searching through $N!$ permutation matrices, we can reduce the original problem to the case when the wanted permutation matrix has the form (D.2).

In fact, if $P \in \mathbb{R}^{N \times N}$ is a permutation matrix, then there exists a permutation $\sigma^P \in \mathcal{S}_N$, such that the entries of P are $P_{i, \sigma_i^P} = 1$ for all $i = 1, \dots, N$, and zero otherwise.

Let \mathcal{P}_{r-k+1} be a subset of the set of all permutation matrices P of size $N \times N$ such that $\sigma_{r-k+2}^P < \sigma_{r-k+3}^P < \dots < \sigma_N^P$. Then, every permutation matrix from $\mathbb{R}^{N \times N}$ can be written in a unique way as a product of two permutation matrices of the form:

$$\begin{bmatrix} I_{r-k+1} & 0 \\ 0 & \Pi' \end{bmatrix} \Delta, \quad (\text{D.23})$$

where $\Delta \in \mathcal{P}_{r-k+1}$.

Thus, our problem becomes to find Π' and Δ such that the columns of $\begin{bmatrix} I_{r-k+1} & 0 \\ 0 & \Pi' \end{bmatrix} \Delta D$ belong to S . So, for each $\Delta \in \mathcal{P}_{r-k+1}$, we can search for a matrix Π' by using our solution to Problem 2. Moreover, if for some Δ there exists a corresponding matrix Π' , then Π' is unique.

So, computationally, we only have to check $|\mathcal{P}_{r-k+1}| = N(N-1) \cdots (N-r+k) \sim N^{r-k+1}$ combinations. Moreover, in practical applications usually $r = k$, thus giving only N possibilities for the matrix Δ through which one should search.

Finally, note that, since for every $\Delta \in \mathcal{P}_{r-k+1}$ there exists at most one Π' such that the matrix (D.23) solves the Problem 1, there are at most $|\mathcal{P}_{r-k+1}|$ permutation matrices that solve the Problem 1.

Relation with a correspondence problem

Our solution to Problem 2 implies, as a corollary, a solution to a classical correspondence problem in a generic case.

Let D be a matrix from (D.1), and let E be some other ordering of the rows of D , i.e.:

$$E = \begin{bmatrix} x_{\pi(1)} & \cdots & x_{\pi(N)} \end{bmatrix}^T, \quad (\text{D.24})$$

for some permutation π . The correspondence problem consists of finding a permutation matrix Π_0 ,

such that

$$\Pi_0 D = E. \tag{D.25}$$

By taking S_E to be a subspace spanned by the columns of E , (D.25) implies that the columns of $\Pi_0 D$ belong to S_E .

In the previous subsection we have obtained the algorithm for solving the problem of finding a permutation matrix Π , such that the columns of ΠD belong to S_E . Also, in a generic case, the rank of S_E is equal to k , and so the number of such permutation matrices Π is at most $|\mathcal{P}_1| = N$. Moreover, by the algorithm from the previous subsection, we can obtain all such permutation matrices and the one that satisfies (D.25) will be the wanted matrix Π_0 .

Appendix E

Subspace Matching: the optimization algorithm

Considering the matching problem, the *Subspace Matching theorem* guarantees the solution's uniqueness on a convex set if we know $r - k + 1$ matches, where r is the model rank and k the data rank, respectively. For the 2D-2D case, the solution's uniqueness is guaranteed when one correspondence is known and we need two matches to apply the theorem in the 3D-2D case. But, in practice, the needed matches can be decreased to $r - k$, as we mentioned in Chapter 3.

As the following expressions show, the model and data can be augmented with synthetic data, satisfying, at same time, a theorem condition (3.13).

$$\mathbf{S}_{aug} = \begin{bmatrix} \mathbf{S}_{synt} & \mathbf{0} \\ \mathbf{0} & \mathbf{S} \end{bmatrix} \mathbf{C} \quad (\text{E.1})$$

$$\mathbf{W}_{aug} = \begin{bmatrix} A_{synt} \mathbf{S}_{synt} & A \overline{\mathbf{W}} \end{bmatrix} \mathbf{C} \quad (\text{E.2})$$

where $\mathbf{C} = \mathbf{I} - \frac{1}{N+N_{synt}} \mathbf{1}_{N+N_{synt}} \mathbf{1}_{N+N_{synt}}^T$ is a *centering* matrix and, A the unknown affine camera model. Matrix A_{synt} , defined by the user, is the synthetic affine transformation, which allows us

to obtain the projections of the synthetic shape \mathbf{S}_{synt} . Note that, \mathbf{S} and \mathbf{S}_{aug} share one dimension. According to (E.1) and (E.2), expression (3.13) is given by

$$\mathbf{W}_{aug}^T \begin{bmatrix} I_{d \times d} & \underline{\mathbf{0}} \\ \underline{\mathbf{0}} & \mathbf{P}_0^T \end{bmatrix} \Pi_{\mathbf{S}_{aug}}^\perp = \underline{\mathbf{0}} \quad (\text{E.3})$$

where $d = (r - k) + N_{synt}$ and \mathbf{P}_0 is the solution of the following problem

$$\begin{aligned} \mathbf{P}^* = \arg \min_{\mathbf{P}} & \quad \left\| \overline{\mathbf{W}}^T \mathbf{P}^T \Pi_{\mathbf{S}_{aug}}^\perp \right\|_F^2 + \frac{c}{2} \|\mathbf{P}\mathbf{1} - \mathbf{1}\|_F^2 + \frac{c}{2} \|\mathbf{P}^T \mathbf{1} - \mathbf{1}\|_F^2 \\ \text{s.t.} & \quad \mathbf{P}_{[1:d:1:d]} = \mathbf{I}_d \\ & \quad p_{ij} \geq 0 \end{aligned} \quad (\text{E.4})$$

Such as the equality constraints of *doubly-stochastic* matrices (3.10,3.11), imposing the identity matrix can be incorporated into the cost function as the expression shows.

$$\begin{aligned} \mathbf{P}^* = \arg \min_{\mathbf{P}} & \quad \left\| \Pi_{\mathbf{S}_b}^\perp \mathbf{P} \mathbf{W}_b^T - b \right\|_F^2 + \frac{c}{2} \|\mathbf{P}\mathbf{1} - \mathbf{1}\|_F^2 + \frac{c}{2} \|\mathbf{P}^T \mathbf{1} - \mathbf{1}\|_F^2 \\ \text{s.t.} & \quad p_{ij} \geq 0 \end{aligned} \quad (\text{E.5})$$

where $b = -\Pi_{\mathbf{S}_a}^\perp \mathbf{W}_a^T$, \mathbf{W}_a and $\Pi_{\mathbf{S}_a}^\perp$ are the first d columns of $\mathbf{W}_{aug} = \begin{bmatrix} \mathbf{W}_a & \mathbf{W}_b \end{bmatrix}$ and $\Pi_{\mathbf{S}_{aug}}^\perp = \begin{bmatrix} \Pi_{\mathbf{S}_a}^\perp & \Pi_{\mathbf{S}_b}^\perp \end{bmatrix}$, respectively. Note that the variable \mathbf{P} is a $[N - (r - k)] \times [N - (r - k)]$ matrix. To simplify the notation and due to its geometric character, we call the cost function of (E.5) by \mathbf{F}_{geo} . The gradient of \mathbf{F}_{geo} is given by

$$\nabla_{\mathbf{P}} \mathbf{F}_{geo} = 2\mathbf{F}\mathbf{P}\mathbf{G} - 2\mathbf{J} + \frac{c}{2} (\mathbf{P}\mathbf{1}\mathbf{1}^T - \mathbf{1}\mathbf{1}^T) + \frac{c}{2} (\mathbf{1}\mathbf{1}^T \mathbf{P} - \mathbf{1}\mathbf{1}^T) \quad (\text{E.6})$$

where $\mathbf{F} = (\Pi_{\mathbf{S}_b}^\perp)^T \Pi_{\mathbf{S}_b}^\perp$, $\mathbf{G} = \mathbf{W}_b^T \mathbf{W}_b$ and $\mathbf{J} = (\Pi_{\mathbf{S}_b}^\perp)^T b \mathbf{W}_b$.

As- discussed in Section 4.2.1, when we are in presence of clutter points, the geometric constraints (E.3) can not be enough to define a unique solution in a convex domain and more information, like photometry, is required. Then, in this case, the used cost function (4.1) and its gradient are given by

$$\begin{aligned} \mathbf{F}_{tot}(\mathbf{P}) &= \left\| \Pi_{\mathbf{S}_c}^\perp \mathbf{P} \mathbf{W}_b^T - b \right\|_F^2 + \frac{c}{2} \|\mathbf{P} \mathbf{1} - \mathbf{1}\|_F^2 \\ &\quad + \frac{c}{2} \|\mathbf{P}^T \mathbf{1} - \mathbf{1}\|_F^2 + \alpha \mathbf{1}^T (\mathcal{C} \odot \mathbf{P}) \mathbf{1} \end{aligned} \quad (\text{E.7})$$

$$\begin{aligned} \nabla_{\mathbf{P}} \mathbf{F}_{tot} &= 2\mathbf{F} \mathbf{P} \mathbf{G} - 2\mathbf{J} + \frac{c}{2} (\mathbf{P} \mathbf{1} \mathbf{1}^T - \mathbf{1} \mathbf{1}^T) \\ &\quad + \frac{c}{2} (\mathbf{1} \mathbf{1}^T \mathbf{P} - \mathbf{1} \mathbf{1}^T) + \alpha \mathcal{C} \end{aligned} \quad (\text{E.8})$$

where $\Pi_{\mathbf{S}_{aug}}^\perp = \begin{bmatrix} \Pi_{\mathbf{S}_a}^\perp & \Pi_{\mathbf{S}_c}^\perp \end{bmatrix}$ and $\Pi_{\mathbf{S}_c}^\perp = \begin{bmatrix} \Pi_{\mathbf{S}_b}^\perp & \underline{\varrho} \end{bmatrix}$. In this case, $\mathbf{F} = (\Pi_{\mathbf{S}_c}^\perp)^T \Pi_{\mathbf{S}_c}^\perp$ and $\mathbf{J} = (\Pi_{\mathbf{S}_c}^\perp)^T b \mathbf{W}_b$. It is important to refer that \mathbf{P} is also a square matrix: if there are n_c clutter points, \mathbf{P} has $[N + n_c - (r - k)] \times [N + n_c - (r - k)]$ entries.

E.1 The Nesterov's projected gradient algorithm

We use the Nesterov's projected gradient to solve our convex problems. This algorithm is given by

In our case, the cost function f is \mathbf{F}_{geo} (E.5) or \mathbf{F}_{tot} (E.8) and projection function $\text{Proj}(X) = \{\forall i, j X_{ij} = \max(X_{ij}, 0)\}$.

The knowledge of Lipschitz constant L , the maximum singular value of the Hessian matrix, is a requirement of the Nesterov's projected gradient. Differentiating the equation (E.6) and using the vectorial formulation, the Hessian matrix is given by

$$H = 2\mathbf{F} \otimes \mathbf{G} + c (\mathbf{1} \mathbf{1}^T \otimes \mathbf{I} + \mathbf{I} \otimes \mathbf{1} \mathbf{1}^T) \quad (\text{E.9})$$

As we referred to earlier, the Hessian matrix has a prohibitive size for these problems, but in our particular case, we do not need to explicitly compute it to calculate its maximum singular value.

The Nesterov's projected gradient($\mathbf{P}, \mathbf{S}^\perp, \mathbf{S}, \overline{\mathbf{W}}, L$)

- Input: Shape spaces $\mathbf{S}, \mathbf{S}^\perp$, Image points $\overline{\mathbf{W}}$, Lipschitz constant L
 Variable: x - random matrix
 Auxiliary variables: $y = x, x_0 = y, k = 1$
 $f(x)$ - Cost function of problem Proj - Projection function
 - while $\|x - \text{Proj}(x - \nabla f(x))\|_2 > T$
 1. $y_0 = y$
 2. $y = \text{Proj}(x - \frac{1}{L}\nabla f(x))$
 3. $x = y + \frac{k-1}{k+2}(y - y_0)$
 4. $k = k+1$
 end
-

Then, L is defined by

$$L = 2\sigma_{max}(\mathbf{W}_b)^2 \sigma_{max}\left(\Pi_{\mathbf{S}_b}^\perp\right)^2 + 2cN \quad (\text{E.10})$$

Note that the feature descriptor term does not change the Hessian matrix H and therefore the Lipschitz constant L because it is a linear term: the Hessian matrix of \mathbf{F}_{tot} is also given by equation E.9.

Bibliography

- [Aanaes 02] H. Aanaes, R. Fisker, K. Astrom & J. M. Carstensen. *Robust Factorization*. IEEE Trans. Pattern Anal. Mach. Intell., vol. 24, no. 9, pages 1215–1225, 2002.
- [Agarwal 09] S. Agarwal, N. Snavely, I. Simon, S. Seitz & R. Szeliski. *Building Rome in a Day*. In ICCV, pages 72–79, 2009.
- [Aguiar 08] P. Aguiar, J. Xavier & M. Stosic. *Spectrally optimal factorization of incomplete matrices*. In The Twenty-First IEEE Conference on Computer Vision and Pattern Recognition (CVPR'08), Anchorage, AK, June 2008.
- [Anandan 02] P. Anandan & M. Irani. *Factorization with Uncertainty*. Int. J. Comput. Vision, vol. 49, pages 101–116, September 2002.
- [Barzilai 88] J. Barzilai & J. Borwein. *Two point step size gradient methods*. IMA Journal of Numerical Analysis, vol. 8, June 1988.
- [Basri 07] R. Basri, D. Jacobs & I. Kemelmacher. *Photometric Stereo with General, Unknown Lighting*. Int. J. Comput. Vision, vol. 72, no. 3, pages 239–257, 2007.
- [Belongie 02] S. Belongie, J. Malik & J. Puzicha. *Shape matching and object recognition using shape contexts*. Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 24, no. 4, pages 509–522, Apr 2002.

- [Berg 05] A. Berg, T. Berg & J. Malik. *Shape matching and object recognition using low distortion correspondences*. Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, vol. 1, pages 26–33 vol. 1, 2005.
- [Bertsekas 99] D. P. Bertsekas. *Nonlinear programming*. Athena Scientific, 2nd edition, 1999.
- [Besl 92] P. Besl & H. McKay. *A method for registration of 3-D shapes*. Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 14, no. 2, pages 239–256, Feb 1992.
- [Breuel 03] T. Breuel. *Implementation techniques for geometric branch-and-bound matching methods*. Computer Vision and Image Understanding, vol. 90, no. 3, pages 258–294, 2003.
- [Buchanan 05] A. Buchanan & A. Fitzgibbon. *Damped Newton Algorithms for Matrix Factorization with Missing Data*. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, volume 2, pages 316–322, 2005.
- [Cai 10] J.-F. Cai, E. Candès & Z. Shen. *A Singular Value Thresholding Algorithm for Matrix Completion*. SIAM Journal on Optimization, vol. 20, pages 1956–1982, March 2010.
- [Candès 11] E. Candès, X. Li, Y. Ma & J. Wright. *Robust Principal Component Analysis?* J. ACM, vol. 58, pages 11:1–11:37, June 2011.
- [Costeira 97] J. Costeira & T. Kanade. *A Multi-body Factorization Method for Independently Moving Objects*. International Journal of Computer Vision, vol. 29, pages 159–179, 1997.
- [David 04] P. David, D. Dementhon, R. Duraiswami & H. Samet. *SoftPOSIT: Simultaneous Pose and Correspondence Determination*. International Journal Computer Vision, vol. 59, no. 3, pages 259–284, 2004.

- [Del Bue 06] A. Del Bue & L. Agapito. *Stereo non-rigid factorization*. International Journal of Computer Vision, vol. 66, no. 2, pages 193–207, February 2006.
- [Del Bue 10] A. Del Bue, J. Xavier, L. Agapito & M. Paladini. *Bilinear Factorization via Augmented Lagrange Multipliers*. In Kostas Daniilidis, Petros Maragos & Nikos Paragios, editeurs, 11th European Conference on Computer Vision (ECCV 2010), Crete, Greece, volume 6314 of *Lecture Notes in Computer Science*, pages 283–296. Springer, 2010.
- [Eriksson 10] A. Eriksson & A. Hengel. *Efficient Computation of Robust Low-Rank Matrix Approximations in the Presence of Missing Data using the L1 Norm*. In The Twenty-Third IEEE Conference on Computer Vision and Pattern Recognition (CVPR'10), San Francisco, CA, June 2010.
- [Ferreira 09] R. Ferreira, J. Xavier & J. Costeira. *Shape From Motion of Nonrigid Objects: The Case of Isometrically Deformable Flat Surfaces*. 2009.
- [Fischler 81] M.A. Fischler & R.C. Bolles. *Random sample consensus: A paradigm for model fitting with application to image analysis and automated cartography*. Communications of the ACM, vol. 24, no. 6, pages 381–395, 1981.
- [Fitzgibbon 03] A. Fitzgibbon. *Robust registration of 2D and 3D point sets*. Image and Vision Computing, vol. 21, no. 13-14, pages 1145–1153, 2003. British Machine Vision Computing 2001.
- [Gold 98] S. Gold, A. Rangarajan, L. Chien-Ping, S. Pappu & E. Mjolsness. *New algorithms for 2D and 3D Point Matching: Pose Estimation and Correspondence*. Pattern Recognition, vol. 31, no. 8, pages 1019–1031, 1998.
- [Golub] G. Golub & Van Loan. *Matrix Computations*.

- [Guerreiro 03] R. Guerreiro & P. Aguiar. *Estimation of Rank Deficient Matrices from Partial Observations: Two-Step Iterative Algorithms*. Proc. Conf. Energy Minimization Methods in Computer Vision and Pattern Recognition., 2003.
- [Guilbert 06] N. Guilbert, A. Bartoli & A. Heyden. *Affine Approximation for Direct Batch Recovery of Euclidean Motion From Sparse Data*. Int. Journal of Computer Vision, vol. 69, no. 3, pages 317–333, September 2006.
- [Hartley 03a] R. Hartley & F. Schaffalitzky. *PowerFactorization: 3D reconstruction with missing or uncertain data*. In Australia-Japan Advanced Workshop on Computer Vision, 2003.
- [Hartley 03b] R. Hartley & A. Zisserman. Multiple view geometry in computer vision. Cambridge University Press, New York, NY, USA, 2 edition, 2003.
- [Heyden 99] A. Heyden, R. Berthilsson & G. Sparr. *An iterative factorization method for projective structure and motion from image sequences*. Image Vision Comput., vol. 17, no. 13, pages 981–991, 1999.
- [Jacobs 97] D. Jacobs. *Linear Fitting with Missing Data: Applications to Structure-from-Motion and to Characterizing Intensity Images*. In IEEE International Conference on Computer Vision, pages 206–212, 1997.
- [Jiang 07] H. Jiang, M. Drew & Z.-N. Li. *Matching by Linear Programming and Successive Convexification*. Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 29, no. 6, pages 959–975, 2007.
- [Kanatani 98] K Kanatani. *Geometric Information Criterion for Model Selection*. Int. J. Computer Vision, vol. 26, no. 3, pages 171–189, 1998.
- [Ke 05] Q. Ke & T. Kanade. *Robust L1 Norm Factorization in the Presence of Outliers and Missing Data by Alternative Convex Programming*. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2005), June 2005.

- [Leordeanu 05] M. Leordeanu & M. Hebert. *A spectral technique for correspondence problems using pairwise constraints*. Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on, vol. 2, pages 1482–1489 Vol. 2, Oct. 2005.
- [Li 07] H. Li & R. Hartley. *The 3D-3D Registration Problem Revisited*. Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on, pages 1–8, Oct. 2007.
- [Lowe 87] D. G. Lowe. *Three-dimensional object recognition from single two-dimensional images*. Artif. Intell., vol. 31, no. 3, pages 355–395, 1987.
- [Lowe 99] D. Lowe. *Object recognition from local scale-invariant features*. IEEE-International Conference on Computer Vision, vol. 2, pages 1150–1157, Sep. 1999.
- [Maciel 03] J. Maciel & J. Costeira. *A global solution to sparse correspondence problems*. TPAMI, vol. 25, no. 2, pages 187–199, Feb 2003.
- [Marques 09] M. Marques, M. Stosic & J. Costeira. *Subspace matching: Unique solution to point matching with geometric constraints*. In ICCV, pages 1288–1294, 2009.
- [Martinec 05] D. Martinec & T. Pajdla. *3D Reconstruction by Fitting Low-rank Matrices with Missing Data*. CVPR, 2005.
- [Minc 88] Henryk Minc. *Nonnegative matrices*. John Wiley&Sons, New York, 1988.
- [Mirsky 75] L. Mirsky. *A trace inequality of John von Neumann*. Monatshefte für Mathematik, vol. 79, pages 303–306, 1975. 10.1007/BF01647331.
- [Morel 09] J.M. Morel & G.S. Yu. *ASIFT: A New Framework For Fully Affine Invariant Image Comparison*. SIAM Journal on Imaging Sciences, vol. 2, no. 2, pages 438–469, 2009.
- [Nesterov 03] Y. Nesterov. *Introductory lectures on convex optimization:a basic course*. Kluwer Academic Publishers, 2003.

- [Oliveira 05] R. Oliveira, J. Costeira & J. Xavier. *Optimal point correspondence through the use of rank constraints*. CVPR, vol. 2, pages 1016–1021 vol. 2, June 2005.
- [Olsson 08] C. Olsson, O. Enqvist & F. Kahl. *A polynomial-time bound for matching and registration with outliers*. Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on, pages 1–8, June 2008.
- [Pires 08] B. Pires, J. F. Moura & J. Xavier. *LASIC: A model invariant framework for correspondence*. ICIP, pages 2356–2359, Oct. 2008.
- [Prados 05] E. Prados & O. Faugeras. *Shape from shading*. In in *Mathematical Models in Computer Vision: The Handbook*. Springer, 2005.
- [Rao 10] S. Rao, R. Tron, R. Vidal & Y. Ma. *Motion Segmentation in the Presence of Outlying, Incomplete, or Corrupted Trajectories*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, pages 1832–1845, 2010.
- [Rodrigues 08] J. J. Rodrigues, P. Aguiar & J. Xavier. *ANSIG - An analytic signature for permutation-invariant two-dimensional shape representation*. CVPR, June 2008.
- [Roweis 97] S. Roweis. *EM algorithms for PCA and SPCA*. Proceedings NIPS, vol. 10, pages 626–632, 1997.
- [Shum 95] H. Shum, K. Ikeuchi & R. Reddy. *Principal component analysis with missing data and its application to polyhedral object modeling*. IEEE Trans. Pattern Anal. Mach. Intell., vol. 9, no. 2, pages 137–154, 1995.
- [Sparr 98] G. Sparr. *Euclidean and Affine Structure/Motion for Uncalibrated Cameras from Affine Shape and Subsidiary Information*. In *SMILE Workshop on Structure from Multiple Images*, Freiburg, 1998.
- [Stosic 11] M. Stosic, M. Marques & J. Costeira. *Convex solution of a permutation problem*. Linear Algebra and its Applications, vol. 434, no. 1, pages 361–369, 2011.

- [Sturm 96] P. Sturm & B. Triggs. *A factorization based algorithm for multi-image projective structure and motion*, 1996.
- [Sudhir 97] G. Sudhir, A. Zisserman & S. Banerjee. *Finding point correspondences in motion sequences preserving affine structure*. Computer Vision and Image Understanding, vol. 68, no. 2, pages 237–246, 1997.
- [Tardif 07] J.-P. Tardif, A. Bartoli, M. Trudeau, N. Guilbert & S. Roy. *Algorithms for Batch Matrix Factorization with Application to Structure-from-Motion*. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, June 2007.
- [Tenenbaum 00] J. B. Tenenbaum & W. T. Freeman. *Separating Style and Content with Bilinear Models*. Neural Computation, vol. 12, no. 6, pages 1247–1283, 2000.
- [Thrun 05] S. Thrun. *Affine structure from sound*. In In NIPS, pages 1353–1360. MIT Press, 2005.
- [Tomasi 92] C. Tomasi & T. Kanade. *Shape and motion from image streams under orthography: a factorization method*. International Journal of Computer Vision, vol. 9, pages 137–154, 1992.
- [Torresani 08] L. Torresani, V. Kolmogorov & C. Rother. *Feature Correspondence Via Graph Matching: Models and Global Optimization*. In ECCV, pages II: 596–609, 2008.
- [Triggs 97] B. Triggs. *Linear projective reconstruction from matching tensors*. Image and Vision Computing, vol. 15, no. 8, pages 617–625, 1997.
- [Turk 91] M. Turk & A. Pentland. *Eigenfaces for recognition*. J. Cognitive Neuroscience, vol. 3, no. 1, pages 71–86, 1991.
- [Veenman 01] C. Veenman, M. Reinders & E. Backer. *Resolving Motion Correspondence for Densely Moving Points*. IEEE Trans. Pattern Anal. Mach. Intell., vol. 23, no. 1, pages 54–72, 2001.

- [Vidal 02] R. Vidal & J. Oliensis. *Structure from Planar Motions with Small Baselines*. In ECCV, pages 383–398, 2002.
- [Vidal 04] R. Vidal & R. Hartley. *Motion segmentation with missing data using PowerFactorization and GPCA*. Proceedings CVPR, vol. 2, pages 310–316, 2004.
- [Wang 09] Z. Wang & H. Xiao. *Dimension-Free Affine Shape Matching Through Subspace Invariance*. IEEE Conference on Computer Vision and Pattern Recognition, June 2009.
- [Wiberg 76] T. Wiberg. *Computation of principal components when data are missing*. Proceedings Symposium of Comp. Stat., pages 229–326, 1976.
- [Wolsey 99] L. Wolsey & G. Nehmouser. *Integer and combinatorial optimization*. John Wiley & Sons, July 1999.
- [Wu 11] L. Wu, A. Ganesh, B. Shi, Y. Matsushita, Y. Wang & Y. Ma. *Convex Optimization Based Low-Rank Matrix Completion and Recovery for Photometric Stereo and Factor Classification*. Submitted to IEEE Transactions on Pattern Analysis and Machine Intelligence, August 2011.