



HAL
open science

Accurate 3D Shape and Displacement Measurement using a Scanning Electron Microscope

Nicolas Cornille

► **To cite this version:**

Nicolas Cornille. Accurate 3D Shape and Displacement Measurement using a Scanning Electron Microscope. Signal and Image processing. INSA de Toulouse, 2005. English. NNT: . tel-00166423

HAL Id: tel-00166423

<https://theses.hal.science/tel-00166423>

Submitted on 6 Aug 2007

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Accurate 3D Shape and Displacement Measurement using a Scanning Electron Microscope

par

Nicolas Cornille

Thèse présentée pour l'obtention du double titre de

Docteur de l'Université
de Caroline du Sud,
Columbia, USA
Spécialité Génie Mécanique

&

Docteur de l'Institut National
des Sciences Appliquées,
Toulouse, France
Spécialité Informatique

Soutenue le 20 juin 2005 devant le jury composé de :

Mr Hugh A. BRUCK	rapporteur	Associate Professor, University of Maryland
M. Jean-Marc LAVEST	rapporteur	Professeur, Université d'Auvergne
M. Michel BORNERT	examinateur	Chercheur, École Polytechnique
M. Michel DEVY	examinateur	Directeur de Recherche, LAAS-CNRS Toulouse
Mr Stephen R. MCNEILL	directeur de thèse	Associate Professor, University of South Carolina
M. Jean-José ORTEU	directeur de thèse	Professeur, École des Mines d'Albi-Carmaux
Mr Michael A. SUTTON	directeur de thèse	Professor, University of South Carolina

Thèse préparée à l'École des Mines d'Albi-Carmaux et à l'Université de Caroline du Sud

Acknowledgments

First, I wish to express my gratitude to my supervisors Dr. Jean-José Orteu, Dr. Stephen R. McNeill and Dr. Michael A. Sutton for their guidance, encouragement and support throughout the three and half years of this work. They offered me the benefit of their experience in experimental mechanics but also gave me the necessary freedom to pursue my own ideas. I wish to thank as well the reviewers Dr. Hugh A. Bruck and Dr. Jean-Marc Lavest and the committee members Dr. Michel Bornert and Dr. Michel Devy for their constructive comments and suggestions to help in the improvement of the dissertation. I am grateful to Dr. Oscar Dillon, former director of the CMS division at NSF, and Dr. Julius Dasch at NASA HQ for their financial support of this work through grants.

Then, I wish to acknowledge Dr. Dorian Garcia who convinces me to enter in the fascinating world of research. I am also grateful to him and Dr. Hubert Schreier for their regular help and the numerous talks we had concerning the computer programming and the computer vision applications, I learned a lot close to them. I am thankful to Dr. Laurent Robert for his advices about deformation measurement and Dr. Ning Li for his help in conducting experiments using the Scanning Electron Microscope. More generally, I would like to thank the staff of Ecole des Mines and of the University of South Carolina and its Electron Microscopy Center. Special thanks to Kay Dorell and Renee Jenkins who always guide me through the paperwork jungle with great kindness.

Finally, I want to thank Alex, Carol, Christine, Didier, Gus, Hicham, Maelle, Olivier, Traian, Vinh and Yann for creating a pleasant work atmosphere and Chan, Christop, Eliette, Eric, Florian, Gilles, Guilhem, Pierre, Vincent for the numerous laughs we had during the coffee breaks not talking about work and the parties not thinking about work. I am grateful to Gwen, Valentina, Francisco and Stephane in USA and Edith, Zabeth, Guillaume and Romain in France for their friendship and the entertaining weekends we spent altogether. I am also thankful to the coffee that kept me awake during this particular period known as “The Writing” where the days last 19 hours and the night only 5, to my very slow computer allowing me to quietly have a coffee break during the compilation time and to the television programs which were so uninteresting that I could concentrate on my work. Most importantly, I wish to express my loving thanks to Sylvie for being by my side

every day for the last 5 years and for the next 100 years. Your everlasting love gave me the courage to face the challenge of this thesis and I greatly appreciated your support and understanding for my work demands.

Abstract

With the current development of nano-technology, there exists an increasing demand for three-dimensional shape and deformation measurements at this reduced-length scale in the field of materials research. Images acquired by Scanning Electron Microscope (SEM) systems coupled with analysis by Digital Image Correlation (DIC) is an interesting combination for development of a high magnification measurement system. However, a SEM is designed for visualization, not for metrological studies, and the application of DIC to the micro- or nano-scale with such a system faces the challenges of calibrating the imaging system and correcting the spatially-varying and time-varying distortions in order to obtain accurate measurements. Moreover, the SEM provides only a single sensor and recovering 3D information is not possible with the classical stereo-vision approach. But the specimen being mounted on the mobile SEM stage, images can be acquired from multiple viewpoints and 3D reconstruction is possible using the principle of videogrammetry for recovering the unknown rigid-body motions undergone by the specimen.

The dissertation emphasizes the new calibration methodology that has been developed because it is a major contribution for the accuracy of 3D shape and deformation measurements at reduced-length scale. It proves that, unlike previous works, image drift and distortion must be taken into account if accurate measurements are to be made with such a system. Necessary background and required theoretical knowledge for the 3D shape measurement using videogrammetry and for in-plane and out-of-plane deformation measurement are presented in details as well. In order to validate our work and demonstrate in particular the obtained measurement accuracy, experimental results resulting from different applications are presented throughout the different chapters. At last, a software gathering different computer vision applications has been developed.

Keywords: Scanning Electron Microscope, Imaging System Calibration, Distortion Correction, Drift Correction, Videogrammetry, 3D Shape Measurement, Strain Measurement, Experimental Mechanics.

Résumé

Avec le développement actuel des nano-technologies, la demande en matière d'étude du comportement des matériaux à des échelles micro ou nanoscopique ne cesse d'augmenter. Pour la mesure de forme ou de déformation tridimensionnelles à ces échelles de grandeur, l'acquisition d'images à partir d'un Microscope Électronique à Balayage (MEB) couplée à l'analyse par corrélation d'images numériques s'est avérée une technique intéressante. Cependant, un MEB est un outil conçu essentiellement pour de la visualisation et son utilisation pour des mesures tridimensionnelles précises pose un certain nombre de difficultés comme par exemple le calibrage du système et la correction des fortes distorsions (spatiales et temporelles) présentes dans les images. De plus, le MEB ne possède qu'un seul capteur et les informations tridimensionnelles souhaitées ne peuvent pas être obtenues par une approche classique de type stéréovision. Cependant, l'échantillon à analyser étant monté sur un support orientable, des images peuvent être acquises sous différents points de vue, ce qui permet une reconstruction tridimensionnelle en utilisant le principe de vidéogrammétrie pour retrouver à partir des seules images les mouvements inconnus du porte-échantillon.

La thèse met l'accent sur la nouvelle technique de calibrage et de correction des distorsions développée car c'est une contribution majeure pour la précision de la mesure de forme et de déformations 3D aux échelles de grandeur étudiées. Elle prouve que, contrairement aux travaux précédents, la prise en compte de la dérive temporelle et des distorsions spatiales d'images est indispensable pour obtenir une précision de mesure suffisante. Les principes permettant la mesure de forme par vidéogrammétrie et le calcul de déformations 2D et 3D sont aussi présentés en détails. Dans le but de valider nos travaux et démontrer en particulier la précision de mesure obtenue, des résultats expérimentaux issus de différentes applications sont présentés tout au long de la thèse. Enfin, un logiciel rassemblant différentes applications de vision par ordinateur a été développé.

Mots-clés : Microscope Électronique à Balayage, Calibrage d'imageurs, Correction de distorsion, Correction de dérive, Vidéogrammétrie, Mesure de forme, Mesure de déformation, Photomécanique.

Contents

Acknowledgments	iii
Abstract	v
List of Figures	xiv
List of Tables	xv
List of Symbols	xvii
1 Problem Statement and Objectives	1
1.1 State of the Art	3
1.2 Thesis Outline	6
Bibliography	8
2 SEM Imaging System: Modeling and Calibration	15
2.1 SEM Imaging Process	16
2.1.1 Principle of Operation of a Scanning Electron Microscope	16
2.1.2 Image Formation	21
2.1.3 Image Disturbances and Distortions	23
2.2 Imaging System Modeling	28
2.2.1 Introduction to Projective Geometry	28
2.2.2 Projection Models	29
2.2.3 Distortion Models	33
2.2.4 Complete Imaging System Model	42
2.3 Imaging System Calibration	44
2.3.1 Calibration Procedure	44
2.3.2 Experimental Process	53
2.4 Experimental Results: From Macro to Nano-Scale	56
2.4.1 Camera Imaging System Calibration	56
2.4.2 Stereo Optical Microscope Imaging System Calibration	58

2.4.3	SEM Imaging System Calibration - Low Magnification	59
2.4.4	SEM Imaging System Calibration - High Magnification	62
	Bibliography	67
3	Automated 3D Reconstruction Process using Videogrammetry	71
3.1	2-View Geometry	74
3.1.1	Epipolar Geometry	75
3.1.2	Fundamental Matrix Estimation	77
3.1.3	Motion Recovery	79
3.1.4	Specific Case of Parallel Projection	81
3.2	Multiple View Geometry	83
3.2.1	3-View Geometry	83
3.2.2	N-View Geometry	85
3.3	Interest Point Detection and Matching	86
3.3.1	Interest Point Detection	86
3.3.2	Interest Point Matching	87
3.4	Triangulation	89
3.4.1	3D Metric Reconstruction from Multi-View Images	89
3.4.2	System Solving: Non-Iterative and Iterative Methods	94
3.4.3	3D Reconstruction using Parallel Projection Imaging Sensors	97
3.5	Applications	100
3.5.1	3D Reconstruction of a Penny Detail	101
3.5.2	3D Reconstruction of a Glass Mold	112
	Bibliography	120
4	Surface Strain Measurement	125
4.1	In-Plane Strain Measurement	126
4.1.1	Strain Tensor Computation	126
4.1.2	Displacement Functions Approximation	129
4.2	Three-dimensional Strain Measurement	134
4.2.1	Experimental Issues due to the SEM Imaging System	135
4.2.2	Strain Tensor Computation	137
4.2.3	Change of Coordinate System	138
4.2.4	Displacement Functions Approximation	139
4.3	Experimental Results	141
4.3.1	Experimental Context: Euro vs. Dollar	141
4.3.2	Calibration	142
4.3.3	3D Reconstruction	142
4.3.4	3D Displacement Measurement	143

4.3.5	Strain Computation	144
	Bibliography	151
5	Software Development	153
5.1	<i>Visiocre</i> Software	154
5.1.1	Modules of Computer Vision	154
5.1.2	Module of Videogrammetry	160
5.2	Libraries	160
5.2.1	Graphic Libraries: <i>Qt</i> and <i>GUILib</i>	160
5.2.2	Scientific Libraries: <i>v3d</i> and <i>csl</i>	164
	Bibliography	166
	Conclusion & Future Work	167
A	Bundle adjustment Implementation	171
	Bibliography	178
B	Introduction to Splines	179
B.1	Two-dimensional Case: Spline Curve	179
B.1.1	Spline Definition	179
B.1.2	B-Spline Definition	180
B.2	Three-dimensional Case: Spline Surface	181
	Bibliography	182
C	Rotations Representations and Conversions	185
C.1	Different Representations	185
C.1.1	Rotation Matrix	185
C.1.2	Rotation Vector	187
C.1.3	Cardan-Bryant Angles Representations	188
C.1.4	Quaternion	188
C.2	Conversions	190
C.2.1	Rotation vector \mapsto Rotation matrix	190
C.2.2	Cardan-Bryant \mapsto Rotation matrix	191
C.2.3	Quaternion \mapsto Rotation matrix	191
C.2.4	Rotation matrix \mapsto Rotation vector	193
C.2.5	Rotation matrix \mapsto Cardan-Bryant	195
C.2.6	Rotation matrix \mapsto Quaternion	196
C.2.7	Rotation vector \mapsto Quaternion	198
C.2.8	Quaternion \mapsto Rotation vector	199

D SEM Calibration - Experimental setup	201
D.1 Preparation	201
D.1.1 SEM Settings	201
D.1.2 Operating Conditions	201
D.2 Experimental Procedure	202
D.3 Example of images	204
General Bibliography	206

List of Figures

2.1	Electron Beam / Specimen Interaction	17
2.2	SE / BSE Signal Comparison	18
2.3	SEM Diagram	20
2.4	Image Formation	21
2.5	Projection Models Depending Upon the Magnification	22
2.6	General Behavior of the Drift	25
2.7	Especially Machined SEM platform	26
2.8	Projective Geometry - Parallel lines	28
2.9	Projective Geometry - Scaling	29
2.10	Projection Model Diagram	31
2.11	Perspective Model	32
2.12	Distortion Model Diagram	34
2.13	Image Vector	37
2.14	Distance in Pixels / Distance in Time	37
2.15	Comparison of the Drift Depending on the Position in the Image	38
2.16	Evolution of the Drift Vector through Time	38
2.17	Global / Local Drift Comparison	40
2.18	Drift Function Model	41
2.19	Complete Imaging System Model	42
2.20	Final Imaging System Model	43
2.21	Bundle adjustment for Calibration	50
2.22	Influence of the S.O.R. Parameter	51
2.23	Image Acquisition Sequence	55
2.24	Speckle Pattern Calibration Target	56
2.25	Standard Circle Calibration Target	57
2.26	Calibration Results - Distortion Magnitude	57
2.27	Speckle Pattern Specimen	58
2.28	Calibration Results - Distortion Correction Field	59
2.29	SEM Calibration Target	60

2.30	Calibration Results - Error Distribution	60
2.31	Calibration Results - Distortion Correction Field	61
2.32	Time Elapsed Between each Image Acquisition	62
2.33	SEM Calibration Target	63
2.34	Experimental Results: Non-Null Disparities	64
2.35	Experimental Results: Distribution of Reprojection Errors	65
2.36	Experimental Results: Reprojection Error Comparison	66
3.1	Multistage Process of the Automated 3D Shape Measurement using Videogram- metry	73
3.2	One-view / Two-view Geometry	74
3.3	Epipolar Geometry	75
3.4	Epipolar Transfer	84
3.5	3D Reconstruction from Multiple Views	90
3.6	Possibilities of Organization of 4 Views	91
3.7	Example of 4 Views All Around an Object	92
3.8	Bundle adjustment for 3D Shape Measurement	97
3.9	Speckle Pattern Image Projection	100
3.10	Penny Detail to Reconstruct in 3D	101
3.11	SEM Calibration Target	102
3.12	Calibration Results - Distortion Correction Field	103
3.13	Reestimated 3D Shape of the Coin	104
3.14	Experimental Setup	105
3.15	Penny Detail Acquisition	106
3.16	First Stages of the Algorithm of Reconstruction	108
3.17	3D Shape of the Penny Detail	109
3.18	Dense Matching	109
3.19	3D Shape of the Penny Detail	110
3.20	3D Shape of the Penny Detail With Texture Mapping	111
3.21	Mold for Pyrex [®] Glass Plate	112
3.22	3D Shape Measurement Comparison	113
3.23	Acquisition of Calibration Images	113
3.24	Calibration Results - Distortion Correction Field	114
3.25	Acquisition of Mold Images	115
3.26	Reconstructed 3D Shape of the Mold	115
3.27	Three Clouds of Points before the Registration and Fusion	116
3.28	Final Cloud of Points Generated by the 3D Laser Scanner	117
3.29	Qualitative Comparison Between the 3D Shapes With and Without Distor- tion Correction	119

4.1	In-Plane Strain	127
4.2	Deformation Gradient Tensor	127
4.3	Point Neighborhood	130
4.4	3D Displacement Measurement	135
4.5	Scale Factor Problem with Three-Dimensional Strain	136
4.6	Three-Dimensional Strain	139
4.7	Coin of 10 Cents of Euro	141
4.8	Micro-Lithography Calibration Target	142
4.9	Calibration Results - Distortion Correction Field	143
4.10	Experimental Setup	144
4.11	Images of the Euro Coin Detail	145
4.12	Reconstructed 3D Shape of the “R” of “EURO”	146
4.13	3D Triangular Mesh	147
4.14	Maximum Principal Strain Values Obtained for the Zero-Deformation Experiment	148
4.15	Minimum Principal Strain Values Obtained for the Zero-Deformation Experiment	149
4.16	Maximum Principal Strain Values Obtained for the Zero-Deformation Experiment	150
5.1	Visiocre Libraries	154
5.2	Module of Primitive Extraction.	155
5.3	Module of 2D Mesh.	156
5.4	Module of Tracking.	157
5.5	Module of Calibration.	158
5.6	Module of Calibration.	159
5.7	Module of Videogrammetry	161
5.8	Module of Videogrammetry	162
5.9	Module of Videogrammetry	163
5.10	Image Visualization Widget	164
A.1	Sparse Structure of the Jacobian Matrix	172
A.2	Hessian Matrix Structure	173
A.3	Hessian Matrix Split	174
A.4	Composition of Block Matrices of the Hessian Matrix	175
A.5	Structure of the Matrix Λ	176
B.1	Knot Distribution	180
B.2	Basis Functions of a Uniform B-Spline	182
B.3	B-Spline Surface	183

List of Tables

- 2.1 Angular Field of View for a Given Magnification 23
- 2.2 Calibration Results - Standard Deviation 58
- 2.3 Calibration Results - Stereo Optical Microscope 59
- 2.4 Calibration Results - SEM Low-Magnification 61

- 4.1 Expressions of Spline Kernels and their Derivatives 133

List of Symbols

s	a scalar (in \mathbb{R})
\mathbf{v}	a vector (in \mathbb{R}^n)
\mathbf{v}^\top	a transposed vector
$\ \mathbf{v}\ _2$	l_2 norm (Euclidean norm) of the vector \mathbf{v}
$\mathbf{0}$	null vector
\mathbf{M}	a matrix
\mathbf{M}^\top	a transposed matrix
$ \mathbf{M} $	determinant of the matrix \mathbf{M}
$\ \mathbf{M}\ $	Frobenius norm of the matrix \mathbf{M}
\mathcal{R}	a coordinate reference system
\mathbf{P}	a 3D point $(X \ Y \ Z)^\top$ in the world coordinate reference system (in \mathbb{R}^3)
$\tilde{\mathbf{P}}$	a 3D point $(\tilde{X} \ \tilde{Y} \ \tilde{Z})^\top$ expressed in homogeneous coordinates (in \mathbb{P}^3)
$\underline{\mathbf{P}}$	a 3D point $(\underline{X} \ \underline{Y} \ \underline{Z})^\top$ in the imaging sensor coordinate system (in \mathbb{R}^3)
$\underline{\mathbf{p}}$	a 2D point $(\underline{x} \ \underline{y})^\top$ in the retinal plane (in \mathbb{R}^2)
\mathbf{p}	a 2D point $(x \ y)^\top$ in the distortion-free image plane (in \mathbb{R}^2)
$\tilde{\mathbf{p}}$	a 2D point $(\tilde{x} \ \tilde{y})^\top$ expressed in homogeneous coordinates (in \mathbb{P}^2)
$\check{\mathbf{p}}$	a spatially-varying distorted 2D image point $(\check{x} \ \check{y})^\top$ (in \mathbb{R}^2)
$\breve{\mathbf{p}}$	a spatially-varying and time-varying distorted 2D image point $(\breve{x} \ \breve{y})^\top$ (in \mathbb{R}^2)
\mathcal{F}	a function
$\frac{\partial \mathcal{F}}{\partial x}$	partial derivative of \mathcal{F} with respect to x

Chapter 1

Problem Statement and Objectives

With the current development of nano-technology, three-dimensional shape and deformation measurements at micro- and nano-scale are more and more required in the field of materials research. Images acquired by Scanning Electron Microscope (SEM) systems coupled with analysis by Digital Image Correlation (DIC) is an interesting combination for development of a high magnification measurement system for the following reasons. First, the SEM offers some undeniable advantages in image acquisition such as its great depth of focus, its possibility of nano-scale resolution, its straightforward use, its magnification range from low ($50\times$) to very high (more than $300000\times$), *etc.* and appears like the perfect tool for this application. Besides, the DIC technique has already proved to be a versatile and effective optical method for measuring three-dimensional shapes and/or deformation at the macro-scale due to its high spatial resolution, its high sensitivity and its non-contacting nature.

However, a SEM is designed for visualization, not for metrological studies, and the image formation is biased by several image disturbances. In order to obtain accurate measurements, a particular attention must be paid to the modeling and calibration of the imaging process and especially to the correction for the underlying distortions. In addition to the common distortion depending on the pixel location in the image (later referred to as “spatially-varying distortion”), another distortion depending on the time occurs with a SEM imaging system: a non-uniform apparent displacement of the specimen across the screen (later referred to as “drift” or “time-varying distortion”).

The application of digital image correlation to the micro- or nano-scale requires that the surface of the observed specimen is random enough (the so-called “speckle pattern” texture) so that DIC can be used to locate and track small sub-regions throughout the image. Therefore, methods must be developed to be able to cover the specimen with an artificial speckle pattern texture at this reduced-length scale if the natural texture cannot be used.

Moreover, the SEM provides only a single sensor and recovering 3D information is not

possible with the classical stereo-vision approach. Acquisition of simultaneous images is indeed not realizable with a single sensor, but the specimen being mounted on a mobile stage, images can be acquired from multiple viewpoints. The unknown rigid-body motions undergone by the specimen can then be recovered *a posteriori* from image analysis and the 3D shape can be measured (Structure-From-Motion method). With the additional objective to study surface deformation of the specimen, three-dimensional displacement are measured as well but due to the uniqueness of the sensor, specific experimental constraints are then necessary.

1.1 State of the Art

For a few decades, computer vision and digital image correlation has been applied to the study of in-plane material behavior at the macro-scale: shape, stress, displacement and deformation analysis [PR82, MNSW⁺83, CRSP85, SCP⁺86]. With this success, DIC technique became one of the preferred optical method of measurements in experimental mechanics and, later, the addition of a second imaging sensor (stereo-vision) allowed to access to the three-dimensional material behavior and to the 3D shape [KJC90, LCSP93, LCS94, HMS96, OGD97, SS99, GH99, SMHS00, Gar01, Sch03]. Nowadays, the 2D and 3D methods using stereo-vision are widely used in many applications and commercial softwares are available [CSI, Tri].

A first step to access a reduced-length scale of measurement was to apply DIC method with images coming from magnifying optical imaging systems, such as the optical microscope for 2D measurements or the stereo-optical microscope for 3D measurements [SCTB90, MDD96, MKWJ99]. However, few authors have investigated the problem of the accurate calibration of the micro-scale imaging systems, and specifically the determination and correction of the underlying distortions in the measurement process. One reason may be that the obvious complexity of high-magnification imaging systems weakens the common underlying assumptions in parametric distortion models (radial, decentering, prismatic, ...) commonly used until then to correct simple lens systems such as digital cameras [Bey92, WCH92, SHL95]. Peuchot [Peu93], Brand *et al.* [BB94] or Schreier *et al.* [SGS04] all proposed to calibrate accurately the imaging sensor by correcting *a priori* for the distortion using a non-parametric model. The *a priori* correction of the distortion transforms the imaging sensor into a virtual distortion-free sensor plane using a calibration target. The same target can then be used to calibrate this ideal virtual imaging sensor using unknown arbitrary motions.

Due to the nature of white light, optical imaging systems are limited to a maximum resolution, leading to a maximum magnification of about 1000 \times . To access smaller scale of measurements, imaging systems based on electron microscopy (such as the TEM and the SEM) are employed. The first issue encountered when using such systems is that the physics of electron microscopy is quite different from optical microscopy and a new model and calibration process is necessary compared to the classical approach. The second issue is that classic electron microscope (such as TEM or SEM) provides only one imaging sensor whereas 3D measurements require at least two images acquired from different viewpoints¹

¹Note that it is actually possible to obtain 3D measurements from only one viewpoint by acquiring images at different zoom levels. This technique is called axial stereo-vision [RA94, SDL96] but it will not be used here.

and then usually use two imaging sensors rigidly mounted on a stereo-rig (stereo-vision).

Concerning the first point, the different projection models have already been established but there is no work today giving a satisfactory model and correction method for the existing image disturbances such as the distortion and the drift. Then, most of the papers and even commercial SEM measurement systems just ignore them and consider a pure projection model [MeX, SAM, HAS⁺96, LTY96, AHD⁺00, VLBB⁺01, SRK⁺02]. The few authors taking into account distortion consider only a parametric model [Dou00, HA00, LSP03] and always neglect the drift effect.

The second point concerning the acquisition of stereo images using a SEM is solved by abandoning the idea of simultaneous images: the two viewpoints are obtained by tilting the specimen between two consecutive acquisitions, under the hypothesis that the specimen shape does not evolve during the time of acquisition of both images [JJ95, LTCY98, RWT00]. However, this is not really equivalent to a stereo-rig because the rigid-body transformation between the two viewpoints is not fixed and cannot be determined by a calibration phase. The motion undergone by the specimen must then be recovered *a posteriori* from image analysis [AN88, HN94, Zha96]. This is possible thanks to a geometric constraint existing between any pair of images: the well-known epipolar geometry [Fau93, LF96], represented by the fundamental matrix. Estimation of epipolar geometry led to a lot of works over the past decade [Tor95, Har95, Zha96, GSV97, TM97, Zha98] and several methods and parameterization of the fundamental matrix exist: iterative or non-iterative method, from 7 or more point correspondences, imposing or ignoring the rank-2 constraint of the fundamental matrix, normalizing or not the data, *etc.* Recently, Armangue proposed a comparison of all the different approaches [AS03].

The natural extension to this technique is the use of multiple views of the specimen, allowing not only a better accuracy in 3D reconstruction but also a complete view of the specimen. This approach is known for macro-scale applications as photogrammetry or videogrammetry [Gru97, Pol99]. The passage from two-view geometry to multi-view geometry leads to new theoretical issues and to the introduction of new objects: the trifocal and quadrifocal tensor introduced in Section 3.2 and discussed in details in [FM95, Lav96, TZ97, Har98, HZ00].

The videogrammetry technique has been recently applied to SEM [CGS⁺03, CGS⁺04, LBVP⁺04] in order to improve in part the quality of 3D shape measurements but the most important point for the accuracy of the 3D reconstruction remains the calibration and the distortion correction. However, the previous works considering spatially-varying distortion in 3D shape measurement always use common parametric models, which are not really adapted to complex imaging systems such as the SEM [CSG⁺04] and do not take into account time-varying distortion.

In addition to the accuracy, a second objective for our 3D reconstruction process is its automation. For this purpose, Section 3.3 introduces interest point detectors which extract automatically feature points in the acquired images. These particular points are very interesting because they are characteristic of an image and can be robustly extracted in an image sequence. They can be matched automatically and allow then to estimate the epipolar geometry without intervention of the user. Numerous interest point detectors exist in the literature: not only the well-known Harris [HS88], SUSAN [SB97] and Canny [Can86] edge detectors, but also Horaud *et al.* [HSV90], Heitger *et al.* [HRVDH⁺92], Cottier [Cot94] and Forstner [For94]. Schmid *et al.* [SMB00] compare them using different criteria to evaluate them.

1.2 Thesis Outline

Chapter 2 presents the preliminary and necessary work required for accurate measurements: the modeling and calibration phase. After a quick introduction to electron microscopy and to the principle of operation of a SEM, the physical aspect of the imaging process is studied to be able to establish the most adequate model. Actually, two different models of projection depending on the magnification and two complementary models of distortions are finally presented.

Calibration is the most difficult and most important step of the whole measurement process because it will determine the accuracy of measurement, particularly for complex imaging system such as the Scanning Electron Microscope. It consists of two different problems: estimation of the projection model parameters and correction for spatially-varying distortion and drift. The former is quickly treated in Chapter 2 because it is equivalent to macro-scale calibration problem and is a known issue in computer vision. The latter leads us to develop a new methodology because previous works always neglect the drift effect and rarely take into account spatially-varying distortion. Moreover, the few authors considering distortion always use a parametric model. Our non-parametric approach is a major contribution of the thesis and is applicable to any imaging system. A comparison of approaches has been made in Section 2.4 and experimental results demonstrate the important accuracy improvement realized with our method. Our new approach of the calibration allows now to transform a simple visualization tool such as the SEM into a real measurement system.

Chapter 3 describes each step of our automated multi-stage approach for 3D reconstruction. It uses a videogrammetry technique because the SEM only provides one sensor and classical 3D shape measurement using stereo-vision cannot be directly applied here. The whole process to recover the unknown rigid-body motions between the acquisitions is first explained for two images and then extended to multiple images. Multi-view allows not only to obtain a better accuracy in the reconstruction but also to have a complete view around the observed specimen.

The last section of this chapter presents experimental results of the 3D reconstruction of a penny detail using the SEM. But the 3D shape measurement approach developed in this work is not specific to reduced-length scale application and can be applied for macro-scale measurements. Another experiment using a sequence of images acquired by a single moving camera is then presented and the accuracy of the method is assessed by comparing the 3D shape measurement with the results obtained using a 3D laser scanner.

Chapter 4 treats the surface deformation measurement from in-plane and out-of-plane displacement measurement using a SEM. The entire process is detailed, from the experimental procedure for measuring 2D and 3D displacement fields to the approximation of

the displacement functions to derive in order to obtain the strains. Limitations due to the SEM imaging system are presented as well and experimental solutions are proposed for the issue due to the single sensor provided by the SEM.

Experimental results are presented at the end of the chapter. In order to validate the entire process of calibration, 3D reconstruction and strain computation, a “zero-deformation” experiment is conducted: two sets of several images of a specimen undergoing only rigid-body motions are acquired and strains (supposed to be null) are computed between the two 3D shapes reconstructed by videogrammetry.

Chapter 5 is dedicated to the different computer programs developed for the needs of this work. In addition to the scientific and graphic libraries, a modular software gathering different computer vision applications and particularly a module of videogrammetry has been developed. This software being aimed at any user, not especially experienced in computer vision, a particular attention has been paid to the user-friendliness (usability) and consequently to the Graphic User Interface.

Bibliography

- [AHD⁺00] M. Agrawal, D. Harwood, R. Duraiswami, L. Davis, and P. Luther. Three Dimensional Ultrastructure from Transmission Electron Microscope Tilt Series. In *Second Indian Conference on Vision, Graphics and Image Processing (ICVGIP 2000)*, Bangalore (India), 2000.
- [AN88] J. Aggarwal and N. Nandhakumar. On the computation of motion from sequences of images – a review. In *Proceedings of IEEE*, volume 76, pages 917–935, 1988.
- [AS03] X. Armangue and J. Salvi. Overall View Regarding Fundamental Matrix Estimation. *Image and Vision Computing*, 21(2):205–220, 2003.
- [BB94] R. Brand, P. Mohr and P. Bobet. Distorsions optiques: corrections dans un modèle projectif. In *Proceedings of 9th Congrès AFCET, Reconnaissances des Formes et Intelligence Artificielle (RFIA'94)*, pages 87–98, Paris (France), Jan 1994.
- [Bey92] Horst A. Beyer. Accurate Calibration of CCD-cameras. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR'92)*, pages 96–101, Urbana Champaign, (IL, USA), Jun 1992.
- [Can86] John F. Canny. A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–714, 1986.
- [CGS⁺03] Nicolas Cornille, Dorian Garcia, Michael A. Sutton, Stephen R. McNeill, and Jean-José Orteu. Automated 3D Reconstruction Using a Scanning Electron Microscope. In *SEM Conference on Experimental and Applied Mechanics*, Charlotte (NC, USA), Jun 2003.
- [CGS⁺04] Nicolas Cornille, Dorian Garcia, Michael A. Sutton, Stephen R. McNeill, and Jean-José Orteu. Initial Experimental Results Using an ESEM for Metrology. In *SEM Conference on Experimental and Applied Mechanics*, Costa Mesa (CA, USA), Jun 2004.
- [Cot94] J.C. Cottier. Extraction et appariements robustes des points d'intérêts de deux images non étalonnées. Technical report, LIFIA-IMAG-INRIA, 1994.
- [CRSP85] T. C. Chu, W. F. Ranson, M. A. Sutton, and W. H. Peters. Application of Digital Image Correlation Techniques to Experimental Mechanics. *Experimental Mechanics*, 25(3):232–244, 1985.

- [CSG⁺04] Nicolas Cornille, Hubert W. Schreier, Dorian Garcia, Michael A. Sutton, Stephen R. McNeill, and Jean-José Orteu. Gridless Calibration of Distorted Visual Sensors. In *Proceedings of Colloque Photomécanique 2004*, Albi (France), 2004.
- [CSI] Vic2D[®] and Vic3D[®], Correlated Solutions Inc., <http://www.correlatedsolutions.com>.
- [Dou00] Pascal Doumalin. *Microextensométrie locale par corrélation d'images numériques*. PhD thesis, Ecole Polytechnique, 2000.
- [Fau93] Olivier Faugeras. *Three-Dimensional Computer Vision: A Geometric Viewpoint*. MIT Press, 1993. ISBN 0-262-06158-9.
- [FM95] O. D. Faugeras and B. Mourrain. On the geometry and algebra of the point and line correspondences between n images. In *Proceedings of the 5th International Conference on Computer Vision (ICCV'95)*, pages 951–962, Boston (MA, USA), 1995.
- [For94] Wolfgang Forstner. A framework for low-level feature extraction. In *Proceedings of the 3rd European Conference on Computer Vision (ECCV'94)*, pages 383–394, Stockholm (Sweden), 1994.
- [Gar01] Dorian Garcia. *Mesure de formes et de champs de déplacements tridimensionnels par stéréo-corrélation d'images*. PhD thesis, Institut National Polytechnique de Toulouse (France), Dec 2001.
- [GH99] K. Galanulis and A. Hofmann. Determination of Forming Limit Diagrams Using an Optical Measurement System. In *7th International Conference on Sheet Metal*, pages 245–252, Erlangen (Germany), Sep 1999.
- [Gru97] Armin Gruen. Fundamentals of videogrammetry: a review. *Human Movement Science*, 16(2-3):155–187, 1997.
- [GSV97] N. Gracias and J. Santos-Victor. Robust estimation of the fundamental matrix and stereo correspondences. In *International Symposium on Intelligent Robotic Systems*, Stockholm (Sweden), Jul 1997.
- [HA00] M. Hemmleb and J. Albertz. Microtopography - The Photogrammetric Determination of Friction Surfaces. In *IAPRS, Vol. XXXIII*, Amsterdam, 2000.
- [Har95] Richard Hartley. In Defence of the 8-point Algorithm. In *5th International Conference on Computer Vision (ICCV'95)*, pages 1064–1070, Boston (MA, USA), Jun 1995.

- [Har98] Richard Hartley. Computation of the Quadrifocal Tensor. In *5th European Conference on Computer Vision (ECCV'98)*, volume 1406, pages 20–35, Freiburg (Germany), Jun 1998.
- [HAS⁺96] M. Hemmleb, J. Albertz, M. Schubert, A. Gleichmann, and J. M. Köhler. Digital microphotogrammetry with the scanning electron microscope. In *International Society for Photogrammetry and Remote Sensing*, 1996.
- [HMS96] J. D. Helm, S. R. McNeill, and M. A. Sutton. Improved 3D Image Correlation for Surface Displacement Measurement. *Optical Engineering*, 35(7):1911–1920, 1996.
- [HN94] T. Huang and A. Netravali. Motion and structure from feature correspondences: a review. In *Proceedings of IEEE*, volume 82, pages 252–268, 1994.
- [HRVDH⁺92] F. Heitger, L. Rosenthaler, R. Von Der Heydt, E. Peterhans, and O. Kuebler. Simulation of neural contour mechanism: from simple to end-stopped cells. *Vision Research*, 32(5):963–981, 1992.
- [HS88] C. Harris and M. Stephens. A Combined Corner and Edge Detector. In *Alvey Vision Conference*, pages 147–151, 1988.
- [HSV90] R. Horaud, T. Skordas, and F. Veillon. Finding geometric and relational structures in an image. In *Proceedings of the 1st European Conference on Computer Vision (ECCV'90)*, pages 374–384, Antibes (France), 1990.
- [HZ00] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.
- [JJ95] Drahomira Janova and Jiri Jan. Robust Surface Reconstruction from Stereo SEM Images. In V. Hlavac and R. Sara, editors, *Computer Analysis of Images and Patterns (CAIP'95)*, pages 900–905, Berlin (Germany), 1995. Springer.
- [KJC90] Z. L. Khan-Jetter and T. C. Chu. Three-Dimensional Displacement Measurements Using Digital Image Correlation and Photogrammic Analysis. *Experimental Mechanics*, 30(1):10–16, 1990.
- [Lav96] Stéphane Laveau. *Géométrie d'un système de n caméras. Théorie, estimation et applications*. PhD thesis, Ecole Polytechnique, May 1996.
- [LBVP⁺04] Guy Le Besnerais, Fabien Vignon, Jean-Louis Pouchou, Denis Boivin, and Pierre Beauchene. Surface Reconstruction from Multiple SEM Images. In

- Workshop on Applications of Computer Vision 2004*, Orlando (FL, USA), Dec 2004.
- [LCS94] P. F. Luo, Y. J. Chao, and M. A. Sutton. Application of Stereo Vision to 3D Deformation Analysis in Fracture Mechanics. *Optical Engineering*, 33:981, 1994.
- [LCSP93] P. F. Luo, Y. J. Chao, M. A. Sutton, and W. H. Peters. Accurate Measurement of Three-Dimensional Deformable and Rigid Bodies Using Computer Vision. *Experimental Mechanics*, 33(2):123–132, 1993.
- [LF96] Q.-T. Luong and O. Faugeras. The fundamental matrix: Theory, algorithms, and stability analysis. *International Journal of Computer Vision*, 17(1):43–76, 1996.
- [LSP03] H.S. Lee, G.H. Shin, and H.D. Park. Digital Surface Modeling for Assessment of Weathering Rate of Weathered Rock in Stone Monuments. In *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume 34, Ancona (Italy), Jul 2003.
- [LTCY98] A.J. Lacey, S. Thacker, S. Crossley, and R.B. Yates. A multi-stage approach to the dense estimation of disparity from stereo SEM images. *Image and Vision Computing*, 16:373–383, 1998.
- [LTY96] A. J. Lacey, N. A. Thacker, and R. B. Yates. Surface Approximation from Industrial SEM Images. In *British Machine Vision Conference (BMVC'96)*, pages 725–734, 1996.
- [MDD96] E. Mazza, G. Danuser, and J. Dual. Light Optical Measurements in Microbars with Nanometer Resolution. *Microsystem Technologies*, 2:83–91, 1996.
- [MeX] MeX[®] software, Alicona Imaging, <http://www.alicon.com/>.
- [MKWJ99] H. L. Mitchell, H. T. Kniest, and O. Won-Jin. Digital Photogrammetry and Microscope Photographs. *Photogrammetric Record*, 16(94):695–704, 1999.
- [MNSW⁺83] S. R. Mc Neill, M. A. Sutton, W. J. Wolters, W. H. Peters, and W. F. Ranson. Determination of Displacements Using an Improved Digital Correlation Method. *Image and Vision Computing*, 1(3):1333–1339, 1983.
- [OGD97] J.-J. Orteu, V. Garric, and M. Devy. Camera Calibration for 3-D Reconstruction: Application to the Measure of 3-D Deformations on Sheet Metal Parts. In *European Symposium on Lasers, Optics and Vision in Manufacturing*, Munich (Germany), Jun 1997.

- [Peu93] Bernard Peuchot. Camera Virtual Equivalent Model – 0.01 Pixel Detector. In *Proceedings of Computerized Medical Imaging and Graphics*, volume 17, pages 289–294, Jul 1993.
- [Pol99] M. Pollefeys. *Self-Calibration and Metric 3D Reconstruction from Uncalibrated Image Sequences*. PhD thesis, Katholieke Universiteit Leuven, May 1999.
- [PR82] W. H. Peters and W. F. Ranson. Digital Imaging Techniques in Experimental Stress Analysis. *Optical Engineering*, 21(3):427–432, 1982.
- [RA94] V. Rodin and A. Ayache. Axial stereovision: modelization and comparison between two calibration methods. In *IEEE International Conference on Image Processing*, Austin (TX, USA), Nov 1994.
- [RWT00] R.G. Richards, M. Wieland, and M. Textor. Advantages of stereo imaging of metallic surfaces with low voltage backscattered electrons in a field emission scanning electron microscope. *Journal of Microscopy*, 199:115–123, Aug 2000.
- [SAM] SAMx[®], 3D_TOPx package, <http://www.samx.com>.
- [SB97] S. M. Smith and J. M. Brady. SUSAN - a new approach to low-level image processing. *International Journal of Computer Vision*, 23(1):45–78, 1997.
- [Sch03] Hubert Schreier. *Investigation of Two and Three-Dimensional Image Correlation Techniques with Applications in Experimental Mechanics*. PhD thesis, University of South Carolina, 2003.
- [SCP⁺86] M.A. Sutton, M. Cheng, W. H. Peters, Y. J. Chao, and S. R. McNeill. Application of an Optimized Digital Correlation Method to Planar Deformation Analysis. *Image and Vision Computing*, 4(3):143–153, 1986.
- [SCTB90] M. A. Sutton, T. L. Chae, J. L. Turner, and H. A. Bruck. Development of a Computer Vision Methodology for the Analysis of Surface Deformations in Magnified Images. In George F. Van Der Voort, editor, *MiCon 90: Advances in Video Technology for Microstructural Control*, ASTM STP 1094, pages 109–132, Philadelphia (PA, USA), 1990. American Society for Testing and Materials.
- [SDL96] P. Sayd, M. Dhome, and J.-M. Lavest. Axial Stereovision for recovering straight cylinders. In *Proceedings of IEEE Conference on Robotics and Cybernetics CESA '96*, pages 560–564, Lille (France), Jul 1996.

- [SGS04] H. Schreier, D. Garcia, and M. A. Sutton. Advances in Light Microscope Stereo Vision. *Experimental Mechanics*, 44(3):278–288, 2004.
- [SHL95] W. Shih, Y-P. Hung, and W-S. Lin. When should we consider lens distortion in camera calibration? *Pattern Recognition*, 28(3):447–461, 1995.
- [SMB00] C. Schmid, R. Mohr, and C. Bauckhage. Evaluation of Interest Point Detectors. *International Journal of Computer Vision*, 37(2):151–172, 2000.
- [SMHS00] M. A. Sutton, S. R. McNeill, J. D. Helm, and H. W. Schreier. Computer Vision Applied to Shape and Deformation Measurement. In *International Conference on Trends in Optical Nondestructive Testing and Inspection*, pages 571–589, Lugano (Switzerland), 2000.
- [SRK⁺02] O. Sinram, M. Ritter, S. Kleindiek, A. Schertel, H. Hohenberg, and J. Albertz. Calibration of an SEM, Using a Nano Positioning Tilting Table and a Microscopic Calibration Pyramid. In *ISPRS Commission V Symposium*, pages 210–215, Corfu (Greece), 2002.
- [SS99] P. Synnergren and M. Sjö Dahl. A Stereoscopic Digital Speckle Photography System for 3D Displacement Field Measurements. *Optics and Lasers in Engineering*, 31:425–443, 1999.
- [TM97] P. H. S. Torr and D. W. Murray. The Development and Comparison of Robust Methods for Estimating the Fundamental Matrix. *International Journal of Computer Vision*, 24(3):271–300, 1997.
- [Tor95] Phil Torr. *Motion Segmentation and Outliers Detection*. PhD thesis, Department of Engineering Science, University of Oxford, 1995.
- [Tri] ARAMIS[®], Trilion Quality Systems, <http://www.trilion.com>.
- [TZ97] P. H. S. Torr and A. Zisserman. Robust Parameterization and Computation of the Trifocal Tensor. *Image and Vision Computing*, 15(8):591–605, Aug 1997.
- [VLBB⁺01] F. Vignon, G. Le Besnerais, D. Boivin, J.-L. Pouchou, and L. Quan. 3D reconstruction from scanning electron microscopy using stereovision and self-calibration. In *Physics in Signal and Image Processing*, Marseille (France), Jan 2001.
- [WCH92] J. Weng, P. Cohen, and M. Herniou. Camera Calibration with Distortion Models and Accuracy Evaluation. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 14, pages 965–980, 1992.

- [Zha96] Zhengyou Zhang. A New Multistage Approach to Motion and Structure Estimation: From Essential Parameters to Euclidean Motion via Fundamental Matrix. Technical Report RR-2910, INRIA, Jun 1996.
- [Zha98] Zhengyou Zhang. A New Multistage Approach to Motion and Structure Estimation by Gradually Enforcing Geometric Constraints. In *Asian Conference on Computer Vision (ACCV'98)*, pages 567–574, Hong Kong, Jan 1998.

Chapter 2

SEM Imaging System: Modeling and Calibration

Metrological studies for quantitative 3D shape and deformation measurements at the micro- and nano-scale using a Scanning Electron Microscope are a very delicate problem. Indeed, the SEM imaging system is not designed for accurate measurements but only for visualization and the critical part of the work is to address different issues such as image distortion and drift. Previous 3D shape or strain measurement using a SEM system [LTCY98, HA00, VLBB⁺01, SRK⁺02] rarely took into account distortion and always neglected the drift problem. For this reason, a new calibration methodology including distortion correction must be developed.

This chapter describes how to model accurately the SEM imaging system and how to calibrate this model. In order to model the most faithfully possible the imaging system, a physical approach of the problem is first investigated. This is the objective of the first section, after a short introduction about the principle of operation of a SEM: understand how the SEM image is produced in order to model the imaging process and interpret the main sources of distortions to be able to correct for them. When the physical meaning is understood, the second section then presents the “mathematical” approach: what kind of projection models for the image formation, what kind of models for distortions? Two different models of projection are given depending on the magnification and the distortion model is finally divided into two complementary models. Section 2.3 describes our original calibration methodology and experimental process and finally, experimental results will be presented in the last section of this chapter in order to validate our approach.

2.1 SEM Imaging Process

2.1.1 Principle of Operation of a Scanning Electron Microscope

Electron Microscopy Introduction

The concept of electron microscopy is born in the 1930's because scientific research required to pass over a limitation of the optical microscopy: its resolution. According to the Rayleigh criterion, the size of the minimum resolvable detail depends directly on the diameter of the aperture and on the wavelength of the source of illumination. In light microscopy, the shortest possible wavelengths are the ultraviolet rays, which set a resolution limit to 200 nm. Since the photon does not allow to go further, the idea was to use another elementary particle with a shorter wavelength: the electron. Compared to the wavelength of light, the wavelength of electron can be up to 100000 times smaller, depending on the accelerating voltage. This is why the SEM has a much greater resolution ability: about 2 nm (atomic-scale resolution). Relative to the resolution, another improvement of electron microscopy compared to light microscopy is the capability of magnification. While the highest magnification attainable with a light microscope is about 1000 times, the highest magnification of an Environmental Scanning Electron Microscope (ESEM) is more than 300000 times. The second limitation of the optical microscopy is its poor depth of field, due to its large aperture angle. On the contrary, in electron microscopy the sample is far from the objective lens compared to the size of the sample and this results in a small aperture angle and therefore, a good depth of field.

The first electron microscope to be developed is the TEM (Transmission Electron Microscope) and it is very similar to its optical counterpart. In both cases, a sample is mounted on a mobile stage and can be observed due to its interaction with a beam created by a source of illumination. This beam passes through a series of lenses to be concentrated and focused on the specimen. Actually, the main difference between the optical and the electron microscope is only that the former uses a light beam whereas the latter is based upon the existing interaction between atoms and an electron beam. The SEM follows the same principle but the sample is analyzed by interpreting different radiations.

Electron Beam / Specimen Interaction

When the electron beam strikes the specimen, incident electrons irradiate the specimen surface, penetrate deeply the material and interact with the electrons orbiting around atoms of the specimen in a volume called "interaction volume¹". This interaction will

¹This affects obviously the spatial resolution of the measurement which is therefore not equal to the size of the incident beam. It is particularly true for X-rays or backscattered electrons which emerge from higher depths.

generate a variety of radiations (see Fig. 2.1) as well as side effects like a local increase of the temperature, induced currents, etc.

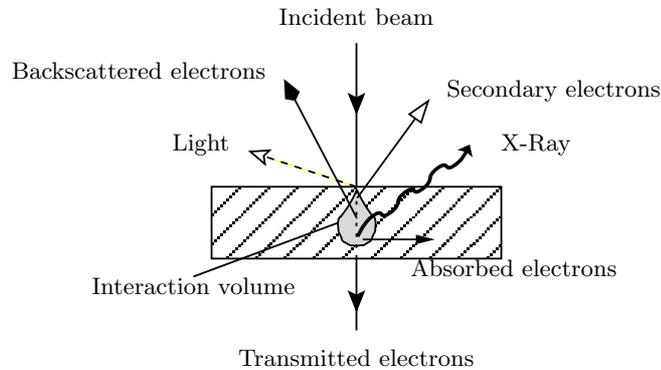


Fig. 2.1 — *Electron beam / Specimen interaction: a variety of signals are emitted when the electron beam hits the specimen.*

All these radiations are produced simultaneously but they are analyzed independently to generate the image, depending on the kind of information desired. Transmitted electrons can only be detected with a TEM and X-ray, backscattered electrons and secondary electrons can be detected using an SEM. But within all these signals, only backscattered and secondary electron emissions are of a particular interest for our work:

- Secondary Electron (SE): they are created by the passage of an incident electron near an atom of the material. The incident electron can transmit a small part of its energy to an electron of this atom which causes its ejection.

Because of their weak energy, only secondary electrons near the surface can emerge from the surface of the specimen and be collected by the detector. Therefore, because the least variation of topography will modify the amount of secondary electrons collected, the information obtained with this detector is mainly correlated to the topography of the specimen.

- Backscattered Electron (BSE): they are created when an incident electron interacts with the nuclei of an atom of the material. Thus, the amount of a backscattered electrons collected depends mostly on the size of the nuclei and then the atomic number.

Contrary to the secondary electrons, the BSE signal represents not only a topography information but also and specifically information about the local specimen atomic composition.

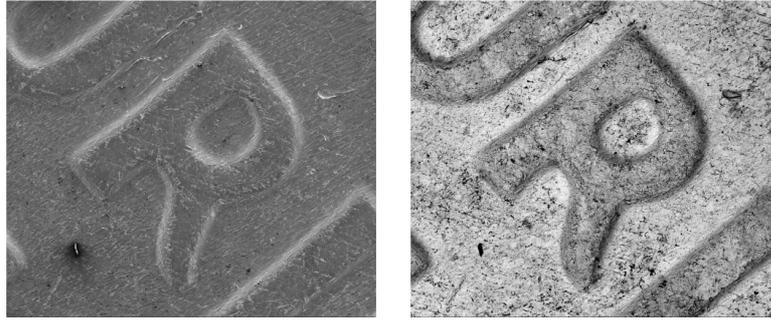


Fig. 2.2 — *Difference between acquisition with SE (left) and BSE detector (right) of a penny detail: the SE detector is strongly influenced by sample topography whereas the BSE is most influenced by local specimen composition.*

SEM General Diagram

For a better understanding of the principle of operation of the SEM, the main components are represented in Fig. 2.3 and are described thereafter:

Electron Gun:

- *Tungsten Filament:* the electron gun generates generally the electron beam from a Tungsten hairpin filament but it can be replaced by Lanthanum Hexaboride (LaB_6) single crystal emitter (which has a longer life and allows more brightness but requires more vacuum). The Tungsten filament acts as a cathode: it is heated by applying a voltage and generates electrons when its temperature attains 2700K.
- *Wehnelt Assembly:* the filament is housed in a Wehnelt assembly which is negatively charged in order to center and concentrate the beam of electrons as the electrons pass through.
- *Anode Plate:* the anode plate is positively charged in order to accelerate electrons. Acceleration voltage varies from about 0.1 kV to about 30 kV. The more the electrons are accelerated, the deeper they will penetrate the specimen.

Vacuum System: vacuum is necessary in the column so that electrons would not be easily deflected by gas molecules in the air. Vacuum system generally consists of two vacuum pumps: one rotary pump and one diffusion pump inside the SEM.

Condenser Lenses: they are electromagnetic lenses and are called like this because they “condense” or demagnify the size of the beam. Reducing the size of the beam allows to control the signal to noise ratio and therefore the resolution. It also reduces the intensity of the beam and then determines the brightness of the image.

Deflection or Scan Coils: deflection coils are regulated by a scan generator and impose a scanning motion on the electron beam. They are involved not only in the image formation but also in the magnification.

Objective Lens: this electromagnetic lens is named “objective” because it is near the object. Its purpose is to focus the beam in a spot on the specimen. It determines the contrast and is involved in the resolution of the image.

SE and BSE Detectors: when the electron beam hits the specimen, a variety of signals are emitted and can be measured by dedicated detectors. The SE detector is generally an Everhart-Thornley detector which attracts secondary electrons by a positive potential. The BSE detector is located just above the sample because it does not attract backscattered electrons but just collects them.

Mobile Stage: the stage is the place where the specimen is loaded. It can be moved in translation (X-axis, Y-axis, Z-axis), rotation (around vertical axis) and tilt (around horizontal axis). By this way, it will be possible to acquire images from different viewpoints and then recover three-dimensional information using the videogrammetry technique (see Section 3).

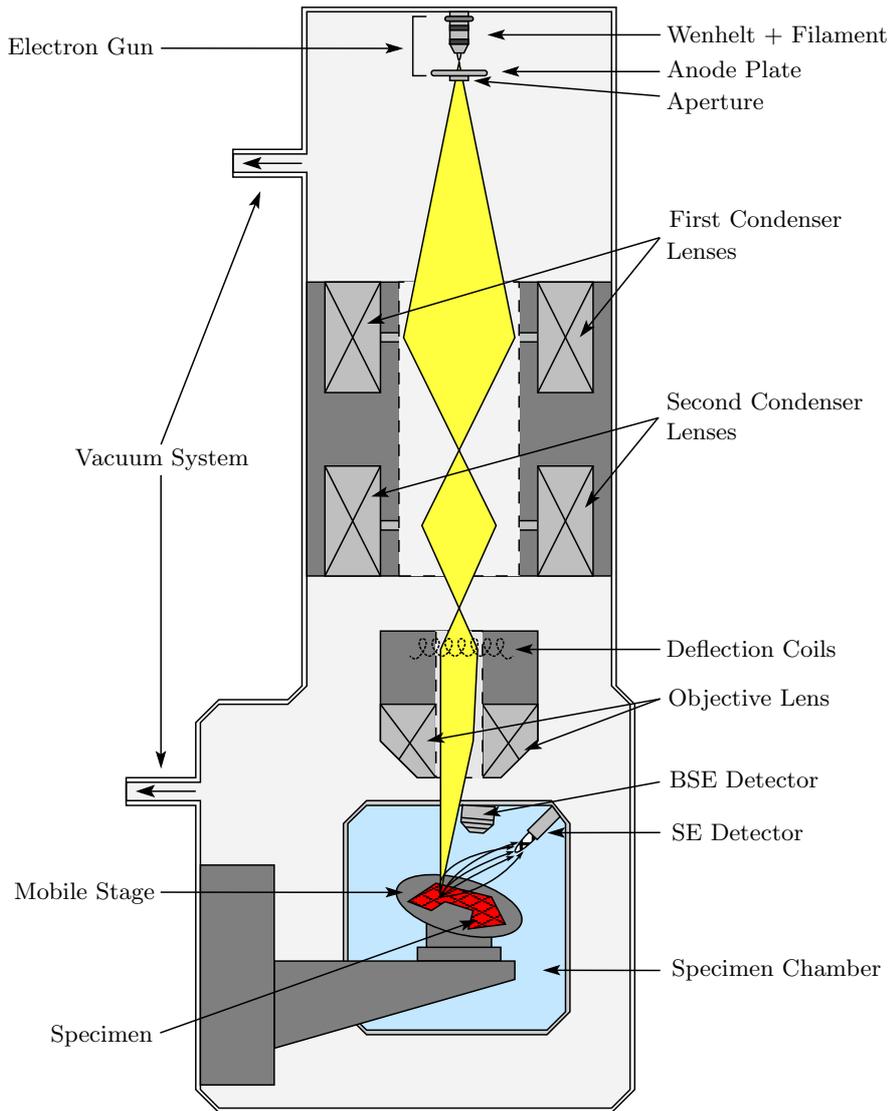


Fig. 2.3 — *SEM diagram.*

2.1.2 Image Formation

The image produced by the SEM is formed by scanning the area of interest of the sample, following a rastering process (see Fig. 2.4). Each pixel of the image is a gray-level representation of the response to the beam excitation of a point (or more exactly of a small area) onto the sample. Therefore, the produced image is a 2D representation of the 3D shape of the sample or in other words, a projected plan of the specimen.

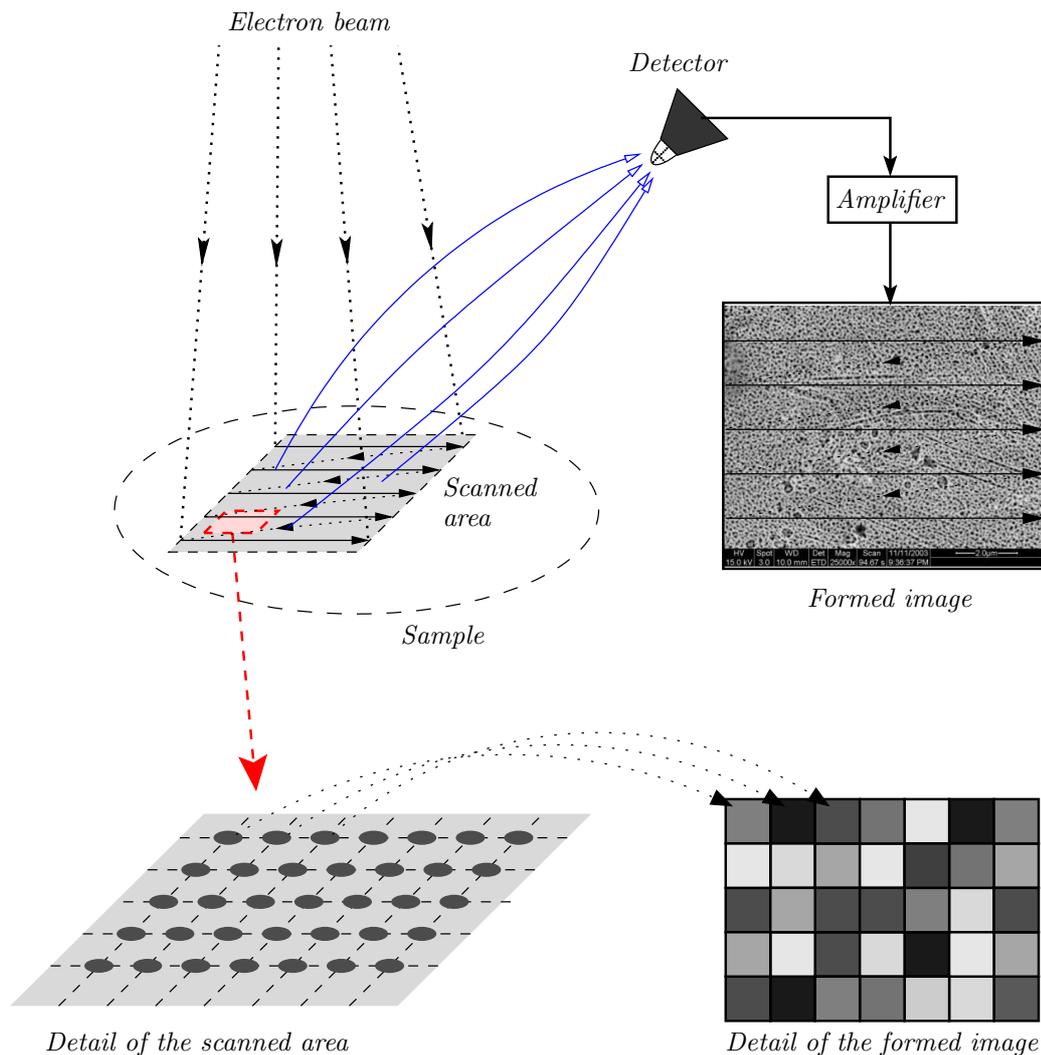


Fig. 2.4 — *Image formation: images of the SEM are formed pixel per pixel following a rastering process (up) where each pixel represents a small area onto the specimen (bottom).*

That is why the SEM imaging process is almost always considered as a pure projective transformation [HAS⁺96, AHD⁺00, VLBB⁺01, SRK⁺02]. Depending on the magnification, two different projection models are used. At low magnification, the general model of

perspective projection² is applied because the field of view (observed area) is large and therefore, the angular field of view (electron beam scanning angle) as well. At higher magnification, the field of view and angular field of view are both very small and projection rays can be considered as parallel: the center of projection is at infinity and the parallel projection is assumed (see Fig. 2.5). Parameters and equations of both projection models are described in detail in Section 2.2.2.

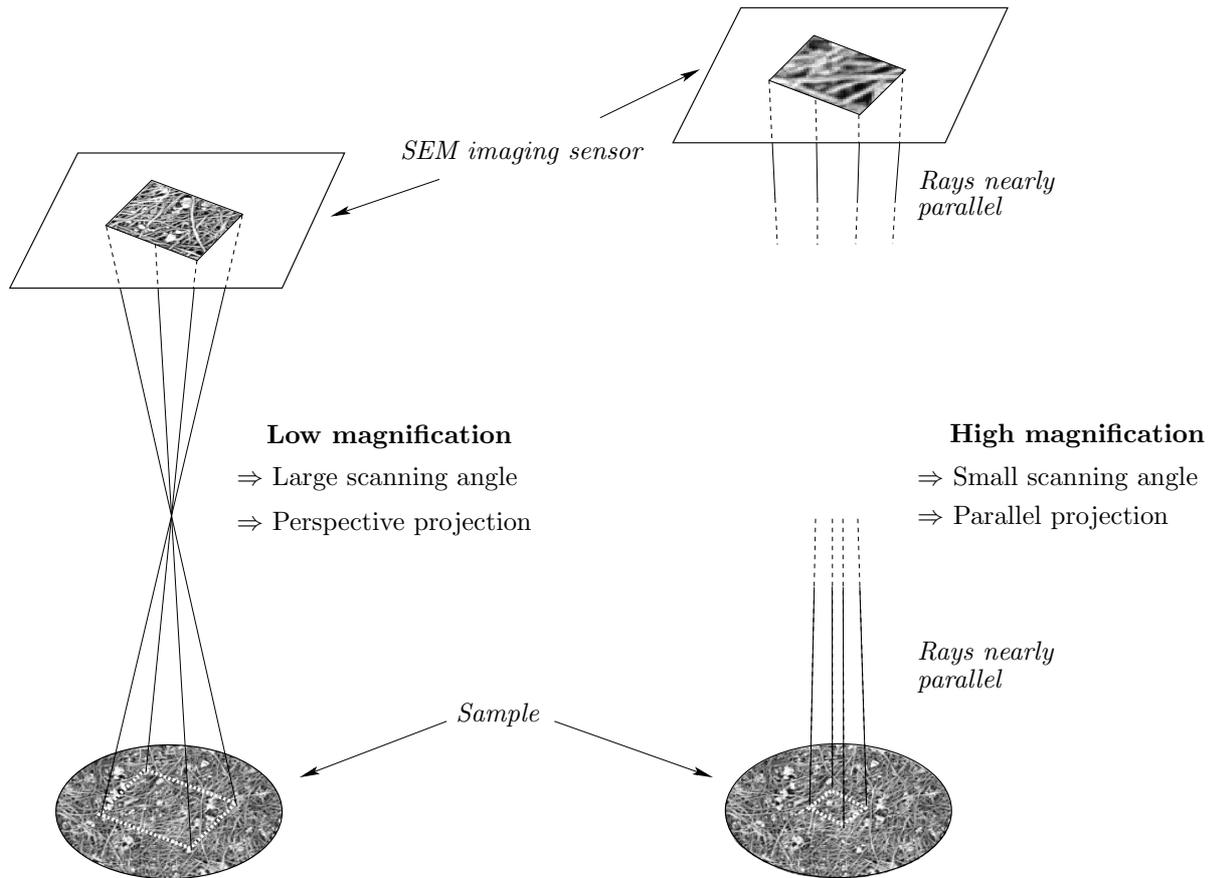
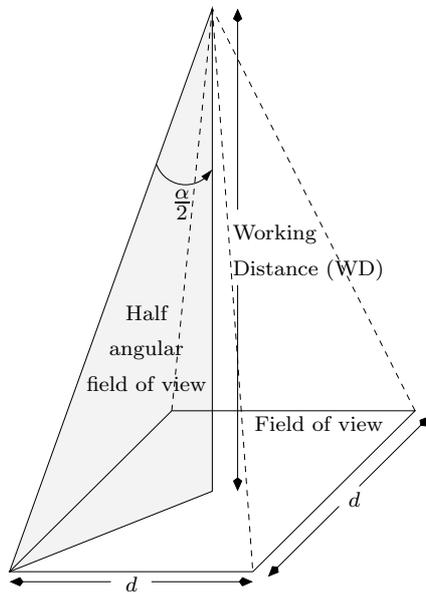


Fig. 2.5 — *Projection models depending upon the magnification: the SEM imaging process is considered as a perspective projection at low magnification and as a parallel projection at high magnification.*

The magnification limit for perspective projection is generally chosen between $200\times$ and $1000\times$ [HA00, VLBB⁺01, SRK⁺02]. However, even over these magnifications, the angular field of view is not negligible and projection rays cannot appropriately be considered as parallel (see Table 2.1). Considering that the angular field of view is negligible only below 0.1° , the approximation of the general perspective projection by the parallel projection

²also known as central projection.

does not seem adequate lower than a magnification of 20000 \times .



Mag.	d (μm)	WD (mm)	α
250 \times	~ 1000	10	$\sim 8.0^\circ$
1000 \times	~ 250	10	$\sim 2.0^\circ$
2500 \times	~ 100	10	$\sim 0.8^\circ$
10000 \times	~ 25	10	$\sim 0.2^\circ$
25000 \times	~ 10	10	$\sim 0.08^\circ$

Table 2.1 — Angular field of view for a given magnification.

In addition to these projection issues, the SEM imaging process is obviously not distortion free. Section 2.1.3 will present different kinds of problem which can cause distortions to take into account and Section 2.2.3 will discuss how to include these distortions in the imaging process model.

2.1.3 Image Disturbances and Distortions

Spatially-varying distortion

In simple lens systems (such as camera or optical microscope), image distortion is a well-known problem. The commonly used method for modeling these imaging systems assumes that such effects result only from lenses aberrations or optical system malfunctions: misalignment of the optical elements, non-parallelism between image plane and sensor plane, lens curvature imperfection, *etc.* In order to correct for these effects, they are approximated by parametric models [Bro71, Bey92, WCH92]: radial distortion, decentering distortion, prismatic distortion, *etc.* and included in the general imaging system model.

The SEM imaging process is based upon the interaction between atoms of the observed specimen and an electron beam. This beam is concentrated and focused using a tray of electromagnetic lenses which are subject to geometrical aberrations and defects such as optical lenses. Therefore, a first approach for modeling the SEM imaging system would consist in considering it as a simple lens system and approximating the image spatially-varying distortions by the parametric models presented earlier. An important limitation

of this approach is that the pre-specified functional forms take into account classical distortions but are ineffective when used to estimate arbitrary lens aberrations or unknown (but deterministic) distortions in the imaging system. In particular, this method should not be adequate for a complex imaging system such as the Scanning Electron Microscope. To address this issue, the work of Schreier *et al.* using a non-parametric distortion correction approach and a speckle-patterned calibration target [SGS04] is used to implement a new methodology of distortion removal from SEM images³. Actually, Peuchot [Peu93] and Brand *et al.* [BB94] first proposed a distortion correction without specific parametric model but it is based upon the use of cross targets, which may be complex to realize at micro- or nano-scale for the SEM. Moreover, a speckle-patterned calibration target leads to a better accuracy of calibration because it allows more measure points for the distortion correction estimation. Parametric and non-parametric methods are both detailed in Section 2.2.3 and the accuracy is assessed in Section 2.4.

Time-varying distortion

An important SEM image disturbance, particularly noticeable at high magnification, is referred to as image drift (non-uniform apparent displacement of the specimen across the screen). This phenomenon has been observed for years but its origin is not really determined and several hypothesis are formulated: deformation due to local heating of the specimen, stabilization problem of the SEM stage due to the controller, accumulation of charges in the SEM column deflecting gradually the electron beam, mechanical vibration. The problem is that the drift obviously occurs during the time of acquisition of one image, and it will result of a displacement of several pixels within the image. Because of its non-stationary nature (its magnitude is function of time), this effect cannot be corrected by the spatially-varying distortion correction procedure and a set of experiments was conducted in order to, first, determine the cause of this effect and then, try to minimize it experimentally.

The following experiments have been realized at a 25000× magnification using a FEI ESEM Quanta 200 with SE imaging:

- (a) standard experiment used for comparisons
- (b) experiment by night and using a small specimen holder in order to minimize the mechanical vibrations
- (c) experiment without using the SEM stage
- (d) experiment with a pause where the beam is off (*i.e.*, the scanning is stopped) during 30 minutes

³This new methodology has also been successfully applied to other imaging systems such as a camera or a stereo-optical microscope [CSG⁺04] (see Section 2.4)

- (e) experiment comparing the rotation of SEM stage and rotation of the beam scanning direction
- (f) qualitative experiment using a SEM from another manufacturer (JEOL SEM)

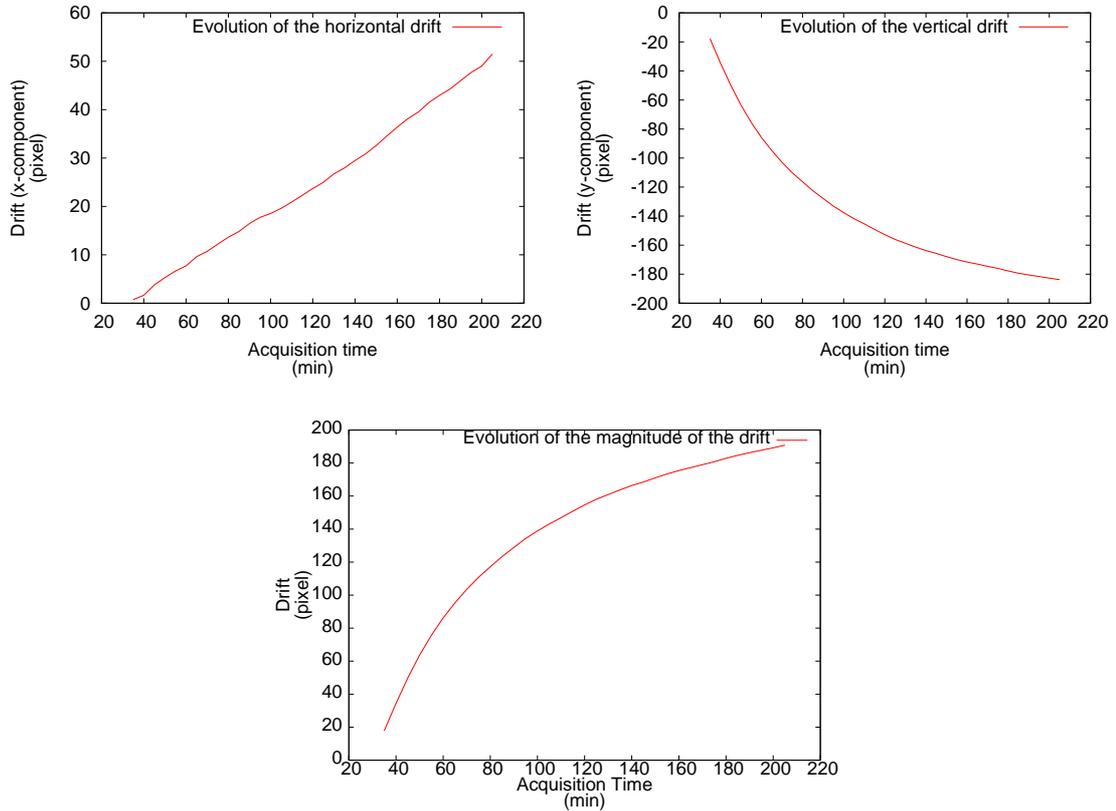


Fig. 2.6 — *General behavior of the mean drift over the image (from left to right: x -component, y -component and magnitude): evolution during 3 hours and 40 minutes.*

The experiment (a) consists in measuring quantitatively the drift in standard operating conditions, so that we can compare the results with following experiments. The experiment lasts 4 hours and consists in acquiring consecutive images each 5 minutes of the same area of a specimen and in the same operating conditions (each image acquisition last about one minute). All images are supposed to be similar (up to the Gaussian noise of measurement) but due to the drift effect, an apparent displacement between them exists. Then, every image is compared by correlation with respect to the first one and the mean disparity (corresponding to the mean displacement, *i.e.* the mean drift over the image) is computed and presented in Fig. 2.6. The direction of the drift changes in each experiment and therefore, only the magnitude will be used for comparison.

Experiment (b) has been conducted using a small specimen holder because bigger ones are more sensitive to vibration and by night to avoid all the possible sources of disturbance

such as people walking in the hall, truck in the street, *etc.* The results are very similar to experiment (a) and allow to think that drift is, for the most part, independent of mechanical vibrations. The experimental process of experiment (c) without using the SEM stage was difficult to accomplish because it required to machine a specific piece (see Fig. 2.7) on which the specimen would be stuck. This platform was placed in the SEM chamber using an available port of the SEM. Because this is more sensible to vibrations than the SEM stage, the results are noisy but the magnitude of the drift is similar to the experiment (a) and we can eliminate the possibility of stabilization of the stage as source of drift. Experiment (d) was conducted to check if the local increase of temperature has an influence on the drift. Indeed, when the electron beam scanned an area, the local temperature increases and may have an influence on the scanning. This is why we stopped the beam during a pause of 30 minutes: if the drift is perturbed by the local temperature, its magnitude when the beam is on again should be less than just before that the beam is off. But results before and after the break show that the drift evolution is not perturbed and is comparable to the evolution of (a), as if the beam was always on. This proves first that there is no relationship between the local increase of temperature and the drift, and secondly that the source of the drift continues to evolve even if the beam is off. Experiment (e) leads to the most impressive results: while a rotation of the SEM stage does not change anything to the drift evolution, a rotation of the beam scanning direction changes the direction of the drift by the same angle. This definitely proves that the drift is dependent of the electron beam scanning and with the results of experiment (d), the hypothesis of the accumulation of charges deflecting the beam seems to be the cause of the drift. No quantitative measures have been realized during experiment (f) but naked-eye observations have been sufficient to clearly see that the drift problem is present as well with a JEOL SEM and is not characteristic to FEI Quanta microscopes.

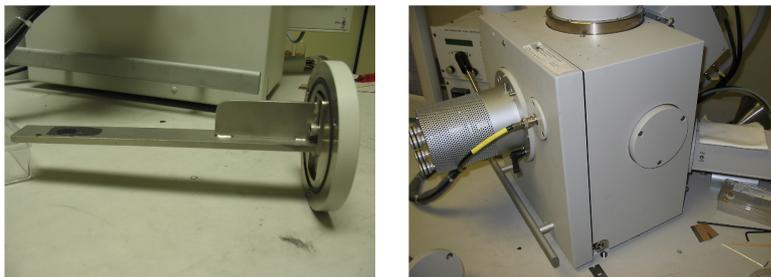


Fig. 2.7 — *Left: platform especially machined for the experiment (c); Right: available port on the side of the SEM used to place the platform.*

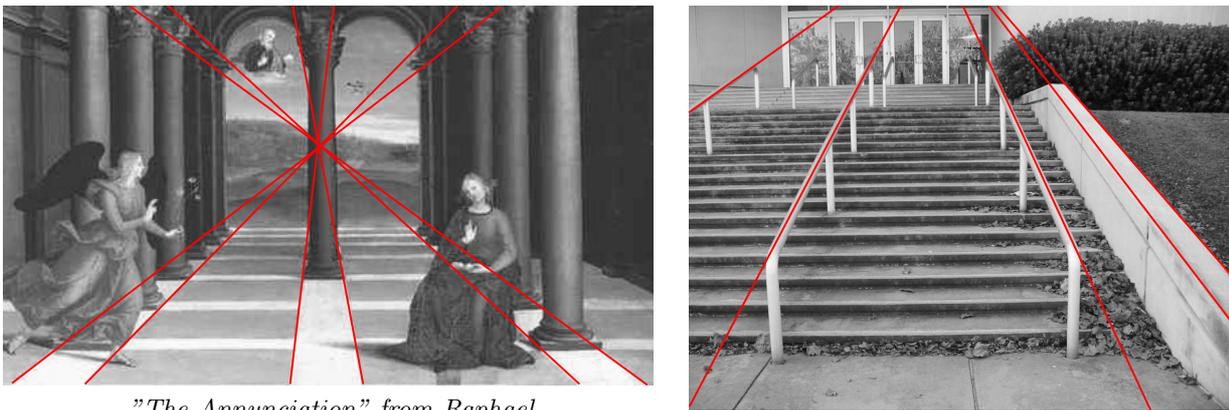
The origin of the drift is now almost certain but unfortunately, there is no obvious and efficient way to minimize it experimentally without adding specific and expensive equipments to the SEM. Based on the behavior of the drift (see Fig. 2.6), one apparent

solution would be to wait several hours before acquiring images, when the drift effect between two successive acquired images can be considered as negligible. But this condition is not acceptable and since the stationary distortion correction procedure is not appropriate for this effect, a specific distortion correction procedure will be developed (see Section 2.2).

2.2 Imaging System Modeling

2.2.1 Introduction to Projective Geometry

It has been shown in the previous section concerning image formation that the SEM imaging system produces a 2D image from a 3D world scene. The geometric transformation that maps 3D world points to 2D image points is called projection and in order to model it, the introduction of a new geometry is necessary: the projective geometry [Fau93]. Indeed, the geometry that describes the world as we can observe and perceive it in three dimensions is the well-known Euclidean geometry but as soon as we want to represent the world on a plane (such as a painter or the imaging process of a camera), Euclidean geometry is inadequate: parallel lines in the world may intersect on the plane (see Fig. 2.8), length ratio is not kept, *etc.*



"The Annunciation" from Raphael

Fig. 2.8 — *Parallel lines in the world (Euclidean space) may not be parallel anymore in a projective space.*

Actually, Euclidean geometry is included in a set of geometries. These geometries are classified and can be considered like strata where the first and simplest is the projective geometry. Then, comes the affine geometry, followed by the similarity or metric geometry and eventually the Euclidean geometry. Only the projective stratum will be briefly presented here in order to introduce the following work where this kind of geometry is widely used (see Chapter 3). For additional information about the notion of stratification, the reader may refer to [DF95, Fau95].

In projective geometry, points in an n -dimensional projective space are defined by a $n+1$ coordinates column vector $\begin{pmatrix} x_1 & x_2 & \dots & x_{n+1} \end{pmatrix}^\top$ such that $\exists i, x_i \neq 0$. $\begin{pmatrix} x_1 & x_2 & \dots & x_{n+1} \end{pmatrix}^\top$ are called homogeneous coordinates of the point. Such points will be denoted \tilde{m} for a 2D point and \tilde{M} for a 3D point in the following sections.

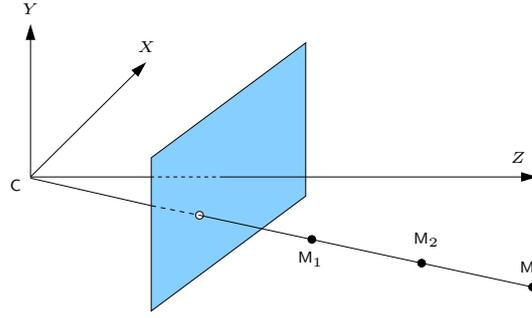


Fig. 2.9 — *In projective geometry, the distance is not important, only the direction is relevant. For instance, the three points M_1 , M_2 and M_3 in Euclidean space will be represented by the same point in a projective space.*

Moreover, scaling is inconsequential in projective geometry (see Fig. 2.9) and two points \tilde{x} and \tilde{y} are said equal (denoted $\tilde{x} \cong \tilde{y}$) if:

$$\exists \lambda \neq 0, \begin{pmatrix} x_1 & x_2 & \dots & x_{n+1} \end{pmatrix}^\top = \lambda \begin{pmatrix} y_1 & y_2 & \dots & y_{n+1} \end{pmatrix}^\top$$

Therefore, a same point can be represented by an infinity of ways and usually, we choose to fix its last coordinate to 1. For example, \tilde{x} would be expressed by this way: $\begin{pmatrix} \frac{x_1}{x_{n+1}} & \frac{x_2}{x_{n+1}} & \dots & 1 \end{pmatrix}^\top$. Obviously, this representation is only available if $x_{n+1} \neq 0$. That's because in projective geometry, a point of which last coordinate is zero corresponds to a point at infinity.

2.2.2 Projection Models

A general projective transformation is a linear transformation mapping any arbitrary 3D world point \tilde{M} in the projective space \mathbb{P}^3 into a 2D point \tilde{m} in the projective space \mathbb{P}^2 . It actually consists of three consecutive transformations:

1. a rigid-body transformation (rotation + translation) \mathbf{T} transforming a 3D point \tilde{M} in the world coordinate system into a 3D point \tilde{M} in the imaging sensor coordinate system:

$$\tilde{M} \cong \mathbf{T}\tilde{M} \tag{2.1}$$

where $\mathbf{T} \cong \begin{pmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^\top & 1 \end{pmatrix}$ with \mathbf{R} the rotation matrix and \mathbf{t} the translation vector.

The parameters of this transformation \mathbf{T} are called extrinsic parameters and represent the orientation and the position of the imaging system in the world.

2. a transformation \mathbf{P} , transforming the 3D point $\underline{\tilde{M}}$ into a 2D point in the retinal plane $\underline{\tilde{m}}$:

$$\underline{\tilde{m}} \cong \mathbf{P} \underline{\tilde{M}} \quad (2.2)$$

This transformation \mathbf{P} mapping a point in \mathbb{P}^3 to a point in \mathbb{P}^2 is the core of the general projective transformation. It is the one that characterizes the perspective or parallel projection for instance.

3. an affine transformation \mathbf{A} , transforming the 2D point $\underline{\tilde{m}}$ into a 2D point \tilde{m} in the distortion-free image plane:

$$\tilde{m} \cong \mathbf{A} \underline{\tilde{m}} \quad (2.3)$$

The parameters of \mathbf{A} are called intrinsic parameters and represent the internal parameters of the imaging sensor.

Composition of the three transformations (2.1), (2.2), (2.3) leads to the following equation for the projective transformation (see Fig. 2.10):

$$\begin{aligned} \tilde{m} &\cong \underbrace{\mathbf{A}\mathbf{P}\mathbf{T}}_{\mathbf{H}} \tilde{M} \\ &\cong \mathbf{H} \tilde{M} \quad \text{where } \tilde{m} \in \mathbb{P}^2, \tilde{M} \in \mathbb{P}^3 \end{aligned} \quad (2.4)$$

where \mathbf{H} is a 3×4 matrix called projection matrix constituted in the most general case of 11 free parameters (and not 12 because in projective geometry, scaling is inconsequential and then the projection matrix is defined up to a scale factor). Note that the two last transformations (2.2) and (2.3) are almost always combined in only one transformation $\mathbf{K} = \mathbf{A}\mathbf{P}$, \mathbf{K} being called intrinsic matrix.

The first special case of projective transformation described thereafter is the perspective projection, modeled by the well-known pinhole model. It will be used for the SEM imaging process at low magnification. The second projective transformation presented is the parallel projection, which can be modeled by 4 different geometric models. The most general case is the affine model, which can be derived in the weak-perspective model. It can be derived itself in the scaled orthographic model, finally derived in the orthographic model. The parallel projection is used at high magnification.

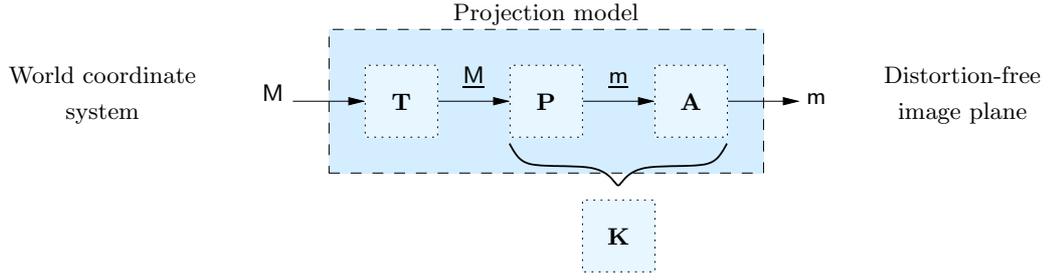


Fig. 2.10 — *Projection model diagram.*

Perspective Projection

The pinhole model [Fau93] is an ideal model of pure perspective projection (see Fig. 2.11) and distortion-free imaging process. Its extrinsic matrix \mathbf{T}_\angle is defined by 6 free parameters (three for the orientation and three for the translational position) and its intrinsic matrix \mathbf{K}_\angle is defined by 5 parameters:

$$\mathbf{A}_\angle = \begin{pmatrix} \alpha_x & s & c_x \\ 0 & \alpha_y & c_y \\ 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad \mathbf{P}_\angle = \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \quad \Rightarrow \quad \mathbf{K}_\angle = \begin{pmatrix} f_x & s & c_x & 0 \\ 0 & f_y & c_y & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

where:

- c_x and c_y are respectively the horizontal and vertical coordinates of the intersection of the optical axis with the image plane, also referred to as principal point. Usually, the principal point is close to the center of the image.
- $f_x = \alpha_x \cdot f$ and $f_y = \alpha_y \cdot f$ are respectively the horizontal and vertical aspect ratio multiplied by the focal length. Usually, the pixels are nearly square and f_x and f_y are nearly equal.
- s is the parameter related to the skew angle ϕ (angle between the x -axis and the y -axis, see Fig. 2.11). Usually, the horizontal and vertical axis are almost perpendicular: the skew angle $\phi \approx 90^\circ$ and $s \approx 0$.

It results in a model with 11 degrees of freedom (5 intrinsic + 6 extrinsic), *i.e.* a projection matrix $\mathbf{H}_\angle = \mathbf{K}_\angle \mathbf{T}_\angle$ with 11 free parameters.

$$\mathbf{T}_\angle = \begin{pmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \Rightarrow \quad \mathbf{H}_\angle \cong \begin{pmatrix} h_{11} & h_{12} & h_{13} & h_{14} \\ h_{21} & h_{22} & h_{23} & h_{24} \\ h_{31} & h_{32} & h_{33} & h_{34} \end{pmatrix}$$

where:

$$\begin{aligned}
 h_{11} &= f_x r_{11} + s r_{21} + c_x r_{31} & h_{12} &= f_x r_{12} + s r_{22} + c_x r_{32} \\
 h_{13} &= f_x r_{13} + s r_{23} + c_x r_{33} & h_{14} &= f_x t_x + s t_y + c_x t_z \\
 h_{21} &= f_y r_{21} + c_y r_{31} & h_{22} &= f_y r_{22} + c_y r_{32} \\
 h_{23} &= f_y r_{23} + c_y r_{33} & h_{24} &= f_y t_y + c_y t_z \\
 h_{31} &= r_{31} & h_{32} &= r_{32} \\
 h_{33} &= r_{33} & h_{34} &= t_z
 \end{aligned}$$

The matrix \mathbf{H}_Z being defined up to a scale, every element is usually divided by t_z to set $h_{34} = 1$.

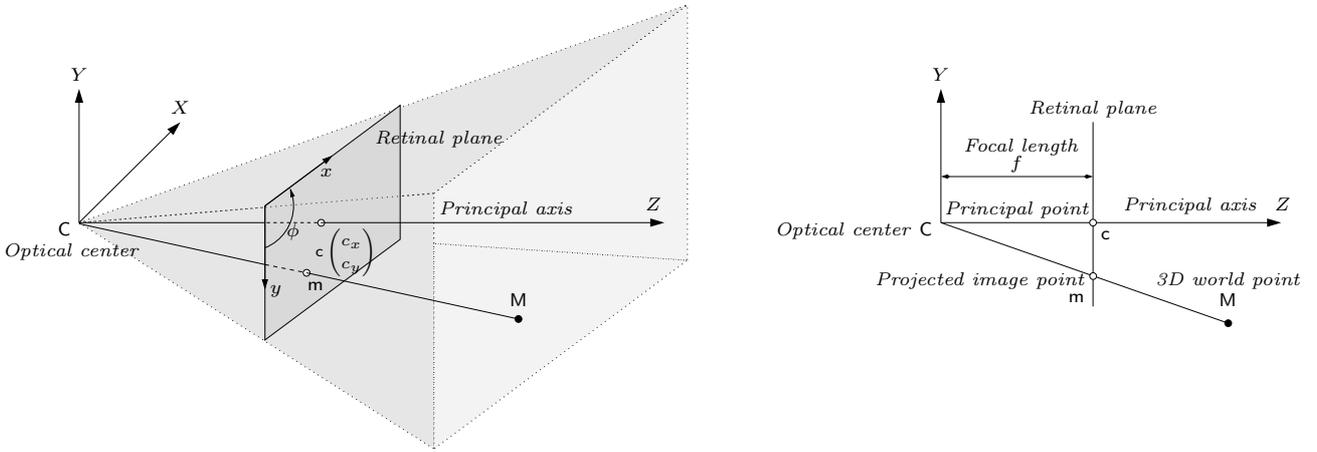


Fig. 2.11 — Perspective model.

Parallel Projection

The parallel projection is a special case of perspective projection where the center of projection would be at infinity. It cannot physically exist but it is assumed when the object is very far from the sensor plane because the angular field of view is very small and projection rays can be considered as parallel. It can be modeled by 4 different geometric models (affine, weak-perspective, scaled orthographic and orthographic model) depending on the conditions made on the internal parameters. For all models, the extrinsic matrix \mathbf{T}_\parallel is the same and is defined by 5 parameters (3 for the orientation and only 2 for the translational position because with parallel projection, the Z-position of the imaging sensor is obviously irrelevant). The general form of \mathbf{K}_\parallel is as follows:

$$\mathbf{A}_\parallel = \begin{pmatrix} \alpha_x & s & 0 \\ 0 & \alpha_y & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad \mathbf{P}_\parallel = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \Rightarrow \quad \mathbf{K}_\parallel = \begin{pmatrix} \alpha_x & s & 0 & 0 \\ 0 & \alpha_y & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

where:

- α_x and α_y are respectively the horizontal and vertical aspect ratio of the image. Both values are different in the affine and the weak-perspective model, equal in the scaled orthographic model and equal to 1 in the orthographic model.
- s is the parameter related to the skew angle. It is non-null only in the affine model.

It results in a projection matrix $\mathbf{H}_{//} = \mathbf{K}_{//}\mathbf{T}_{//}$ with 8 degrees of freedom (3 intrinsic + 5 extrinsic parameters) for the affine model, 7 (2 + 5) for the weak-perspective model, 6 (1 + 5) for the scaled orthographic and 5 (0 + 5) for the orthographic model.

$$\mathbf{T}_{//} = \begin{pmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ * & * & * & * \\ 0 & 0 & 0 & 1 \end{pmatrix} \Rightarrow \mathbf{H}_{//} \cong \begin{pmatrix} h_{11} & h_{12} & h_{13} & h_{14} \\ h_{21} & h_{22} & h_{23} & h_{24} \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

where:

$$\begin{aligned} h_{11} &= \alpha_x r_{11} + s r_{21} & h_{12} &= \alpha_x r_{12} + s r_{22} \\ h_{13} &= \alpha_x r_{13} + s r_{23} & h_{14} &= \alpha_x t_x + s t_y \\ h_{21} &= \alpha_y r_{21} & h_{22} &= \alpha_y r_{22} \\ h_{23} &= \alpha_y r_{23} & h_{24} &= \alpha_y t_y \end{aligned}$$

2.2.3 Distortion Models

For applications that require accurate measurements, a complex imaging system such as the SEM cannot be represented only by a linear model (pure projection model), it requires additional non-linear functional forms to take into account distortions. They are expressed as a final transformation, applied after the projective transformation⁴, and mapping the ideal distortion-free image point \mathbf{m} into a real distorted image point $\check{\mathbf{m}}$:

$$\check{\mathbf{m}} = \mathcal{D}(\mathbf{m}) = (\mathcal{D}_t \circ \mathcal{D}_s)(\mathbf{m}) \quad (2.5)$$

where \mathcal{D} is the general distortion function, composition of both types of distortion: time-varying distortion represented by the function \mathcal{D}_t and spatially-varying distortion represented by \mathcal{D}_s . The whole distortion model is illustrated in Fig. 2.12.

⁴Physically, it should seem more pertinent to place the distortion transformation in the retinal plane, before the affine transformation \mathbf{A} (see Fig. 2.10), but it is mathematically equivalent to place it after and it is more convenient for calibration.

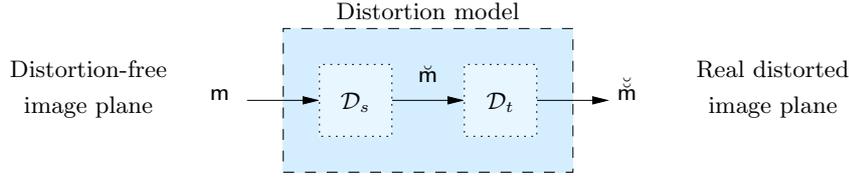


Fig. 2.12 — *Distortion Model Diagram.*

Spatially-Varying Distortion Model

Parametric model: It has been said that spatially-varying distortion is a common problem encountered with all imaging system and not only the SEM. For simple lens systems such as a camera, many authors have proved that parametric models of distortion lead to satisfactory results [SHL95] and that's why common models such as radial, tangential and prismatic distortion model will be used in a first approach.

The spatially-varying distortion function is usually expressed as the sum of the original point \mathbf{m} and an additive distortion term δ_s , itself composed of the three pre-specified elementary distortions:

$$\begin{aligned} \check{\mathbf{m}} = \mathcal{D}_s(\mathbf{m}) &= \mathbf{m} + \delta_s(\mathbf{m}) \\ &= \mathbf{m} + \underbrace{\delta_r(\mathbf{m}) + \delta_d(\mathbf{m}) + \delta_p(\mathbf{m})}_{\substack{\text{radial} \quad \text{decentering} \quad \text{prismatic}}} \end{aligned}$$

The distortion functions δ_r , δ_d and δ_p are usually expressed in a polar coordinate system. Let's note ρ the distance from the point \mathbf{m} to the principal point \mathbf{c} and θ the angle between the x -axis and $\vec{\mathbf{cm}}$. The point \mathbf{m} is then expressed as a function of ρ and θ :

$$\mathbf{m} = \begin{pmatrix} x \\ y \end{pmatrix} = \rho \begin{pmatrix} \cos \theta \\ \sin \theta \end{pmatrix} \quad (2.6)$$

and each distortion function δ_* is expressed with a radial component δ_*^{rad} and a tangential component δ_*^{tan} :

$$\delta_*(\mathbf{m}) = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} \delta_*^{rad}(\rho, \theta) \\ \delta_*^{tan}(\rho, \theta) \end{pmatrix} \quad (2.7)$$

The expression of the radial and tangential components of the three distortion functions is given by Weng [WCH92]:

$$\begin{cases} \delta_r^{rad}(\rho, \theta) = \sum_{n=1}^{\infty} k_n \rho^{2n+1} = k_1 \rho^3 + k_2 \rho^5 + k_3 \rho^7 + \dots \\ \delta_r^{tan}(\rho, \theta) = 0 \end{cases} \quad (2.8)$$

$$\begin{cases} \delta_d^{rad}(\rho, \theta) = 3 \sin(\theta - \theta_d) \sum_{n=1}^{\infty} j_n \rho^{2n} = 3 \sin(\theta - \theta_d)(j_1 \rho^2 + j_2 \rho^4 + j_3 \rho^6 + \dots) \\ \delta_d^{tan}(\rho, \theta) = \cos(\theta - \theta_d) \sum_{n=1}^{\infty} j_n \rho^{2n} = \cos(\theta - \theta_d)(j_1 \rho^2 + j_2 \rho^4 + j_3 \rho^6 + \dots) \end{cases} \quad (2.9)$$

$$\begin{cases} \delta_p^{rad}(\rho, \theta) = \sin(\theta - \theta_p) \sum_{n=1}^{\infty} i_n \rho^{2n} = \sin(\theta - \theta_p)(i_1 \rho^2 + i_2 \rho^4 + i_3 \rho^6 + \dots) \\ \delta_p^{tan}(\rho, \theta) = \cos(\theta - \theta_p) \sum_{n=1}^{\infty} i_n \rho^{2n} = \cos(\theta - \theta_p)(i_1 \rho^2 + i_2 \rho^4 + i_3 \rho^6 + \dots) \end{cases} \quad (2.10)$$

where θ_d and θ_p are the angles between the x -axis and the axis of maximum tangential distortion respectively for the decentering and the prismatic distortion.

Radial distortion being the main component of distortion, (2.8) is limited to the third term $n = 3$ whereas decentering and prismatic distortion are minor distortions and terms of (2.9) and (2.10) of order higher than 2 (corresponding to $n = 1$) are assumed negligible. Using Equ. (2.6) and Equ. (2.7), the three distortion functions (2.8), (2.9) and (2.10) can be expressed in Cartesian coordinates:

$$\begin{aligned} \delta_r(\mathbf{m}) &= (k_1(x^2 + y^2) + k_2(x^2 + y^2)^2 + k_3(x^2 + y^2)^3) \mathbf{m} \\ \delta_d(\mathbf{m}) &= j_1 \begin{pmatrix} -\sin \theta_d(3x^2 + y^2) + 2 \cos \theta_d xy \\ -2 \sin \theta_d xy + \cos \theta_d(x^2 + 3y^2) \end{pmatrix} \\ \delta_p(\mathbf{m}) &= i_1(x^2 + y^2) \begin{pmatrix} -\sin \theta_p \\ \cos \theta_p \end{pmatrix} \end{aligned}$$

Note that $\delta_r(\mathbf{m}, k_1, k_2, k_3)$, $\delta_d(\mathbf{m}, j_1, \theta_d)$, $\delta_p(\mathbf{m}, i_1, \theta_p)$ are respectively denoted $\delta_r(\mathbf{m})$, $\delta_d(\mathbf{m})$, $\delta_p(\mathbf{m})$ for convenience.

Let's note $d_1 = -j_1 \sin \theta_d$, $d_2 = j_1 \cos \theta_d$, $p_1 = -i_1 \sin \theta_p$ and $p_2 = i_1 \cos \theta_p$ in order to simplify the equations and obtain:

$$\begin{aligned} \delta_s(\mathbf{m}) &= \delta_r(\mathbf{m}) + \delta_d(\mathbf{m}) + \delta_p(\mathbf{m}) \\ &= \mathbf{m} (k_1(x^2 + y^2) + k_2(x^2 + y^2)^2 + k_3(x^2 + y^2)^3) \\ &\quad + \begin{pmatrix} d_1(3x^2 + y^2) + 2d_2xy + p_1(x^2 + y^2) \\ 2d_1xy + d_2(x^2 + 3y^2) + p_2(x^2 + y^2) \end{pmatrix} \end{aligned}$$

leading at last to the following expression for the total spatially-varying distortion function:

$$\begin{aligned} \mathcal{D}_s(\mathbf{m}, \mathbf{d}) = & \mathbf{m} \left(1 + k_1(x^2 + y^2) + k_2(x^2 + y^2)^2 + k_3(x^2 + y^2)^3 \right) \\ & + \begin{pmatrix} d_1(3x^2 + y^2) + 2d_2xy + p_1(x^2 + y^2) \\ 2d_1xy + d_2(x^2 + 3y^2) + p_2(x^2 + y^2) \end{pmatrix} \end{aligned}$$

where $\mathbf{d} = (k_1 \ k_2 \ k_3 \ d_1 \ d_2 \ p_1 \ p_2)$ is the vector of distortion parameters.

Non-parametric model: Recently, based on the work of Schreier et al. [SGS04] about the use of spline functions to remove spatially-varying distortion, we implemented a novel distortion correction methodology [CSG⁺04] for SEM images. The idea is to recover the distortion shape from the remaining reprojection errors of the calibration of a pure projective model and fitting it with spline functions that allow to represent any type of distortion field, even complex. The spline fitting and the constraints to insure the uniqueness of its solution are discussed in Section 2.3.

Then, a second possible model for spatially-varying distortion is a spline surface representation:

$$\mathcal{D}_s(\mathbf{m}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \left(\sum_{i=0}^{n_i} \sum_{j=0}^{n_j} \alpha_{i,j} N_i(x) N_j(y) \quad \sum_{i=0}^{n_i} \sum_{j=0}^{n_j} \beta_{i,j} N_i(x) N_j(y) \right)^\top \quad (2.11)$$

where $\boldsymbol{\alpha} = (\alpha_{0,0} \ \dots \ \alpha_{0,j} \ \dots \ \alpha_{0,n_j} \ \alpha_{1,0} \ \dots \ \alpha_{n_i,n_j})^\top$ and $\boldsymbol{\beta}$ are the spline coefficients vectors, N_i and N_j the basis functions and n_i and n_j are equal to the degree of the spline plus respectively the horizontal and vertical number of patches (see Appendix B for an introduction to splines).

Time-Varying Distortion Model

As described in Section 2.1.2 and Fig. 2.4, SEM images are produced pixel by pixel following a rastering process. Each pixel requires an amount of time t_d called “dwell time” to be acquired and a line is acquired in $t_l = w.t_d + t_j$ where w is the image width (number of pixels horizontally) and t_j the delay for repositioning the beam and stabilizing it to the next line. An entire frame is acquired in $t_f = (h - 1).t_l + w.t_d$ where h is the image height (number of pixels vertically).

To make easier the study of the time-varying distortion, the intensity of the image is here considered as a function of time $\mathcal{I}(t)$ and not anymore as a function of image position $\mathcal{I}(\check{x}, \check{y})$. The relationship between time and position is: $t = \check{x}.t_d + \check{y}.t_l$, $0 \leq \check{x} \leq w - 1$, $0 \leq$

$\check{y} \leq h - 1$. Therefore, the image is no longer represented as a 2D frame but as a 1D vector (see Fig. 2.13).

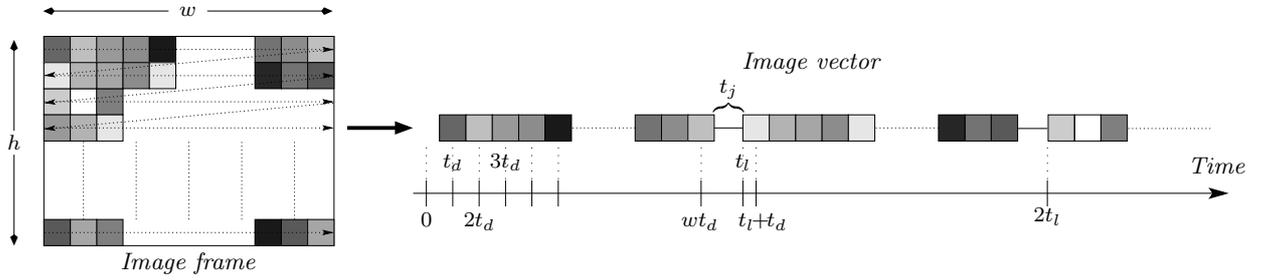


Fig. 2.13 — When the image is a function of time, it is represented as a vector instead of as a frame.

In order to prove that the drift is indeed a function of time and does not depend on the spatial position of the point, four points in the first image of the experiment (a) (previously seen in Section 2.1.3) have been tracked through time over the image sequence and the behaviors of their corresponding drift are compared. We remind that as results of this experiment, the evolution of the drift was presented in Fig. 2.6. The four chosen points are at coordinates $\mathbf{p}_1 = (200 \ 200)$, $\mathbf{p}_2 = (800 \ 200)$, $\mathbf{p}_3 = (200 \ 800)$ and $\mathbf{p}_4 = (800 \ 800)$. Fig. 2.14 illustrates the fact that due to the rastering process (see Fig. 2.4), even the distance between \mathbf{p}_1 and \mathbf{p}_2 and the distance between \mathbf{p}_2 and \mathbf{p}_4 are both equal to 600 pixels (in image frame), the acquisition time between \mathbf{p}_1 and \mathbf{p}_2 is a lot smaller than the acquisition time between \mathbf{p}_2 and \mathbf{p}_4 (in image vector) because \mathbf{p}_1 and \mathbf{p}_2 are on the same row.

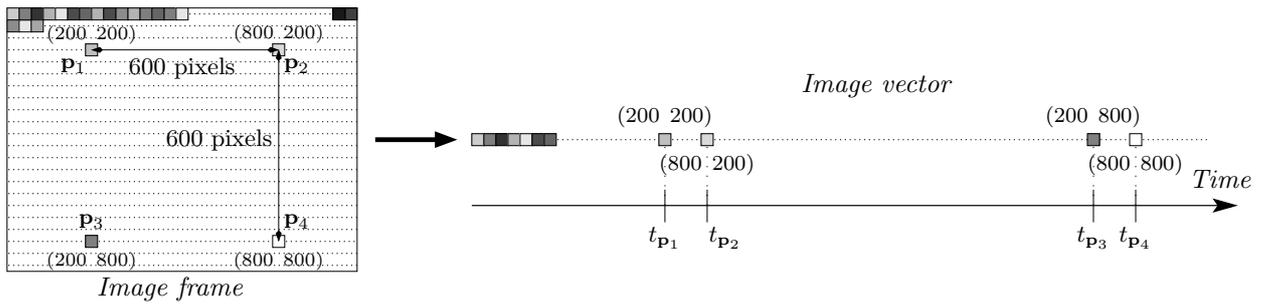


Fig. 2.14 — Distance in pixels (in image frame) is not related to distance in time (in image vector): \mathbf{p}_1 and \mathbf{p}_2 or \mathbf{p}_2 and \mathbf{p}_4 are both separated by 600 pixels but the acquisition time between \mathbf{p}_1 and \mathbf{p}_2 is a lot smaller than the acquisition time between \mathbf{p}_2 and \mathbf{p}_4 .

Comparison of the drift depending on the position in the image is represented in Fig. 2.15 where evolutions of the horizontal and vertical drift for the four chosen points are plotted:

drift evolutions are very similar (graphs seem merged) whatever the position in the image, which is perfectly consistent with the fact that the drift is a time-varying and not a spatially-varying distortion. We can also notice that the only observable difference occurs between the couple of points \mathbf{p}_1 and \mathbf{p}_2 and the couple of points \mathbf{p}_3 and \mathbf{p}_4 (see Fig. 2.15, right). This is because \mathbf{p}_1 and \mathbf{p}_3 are separated by 600 rows and therefore, the time elapsed between their acquisition is very long compared to the time elapsed between the acquisition of \mathbf{p}_1 and \mathbf{p}_2 , separated by 600 columns but on the same row (see Fig. 2.14). Tracking (matching through time) of the points \mathbf{p}_1 and \mathbf{p}_2 is represented in Fig. 2.16.

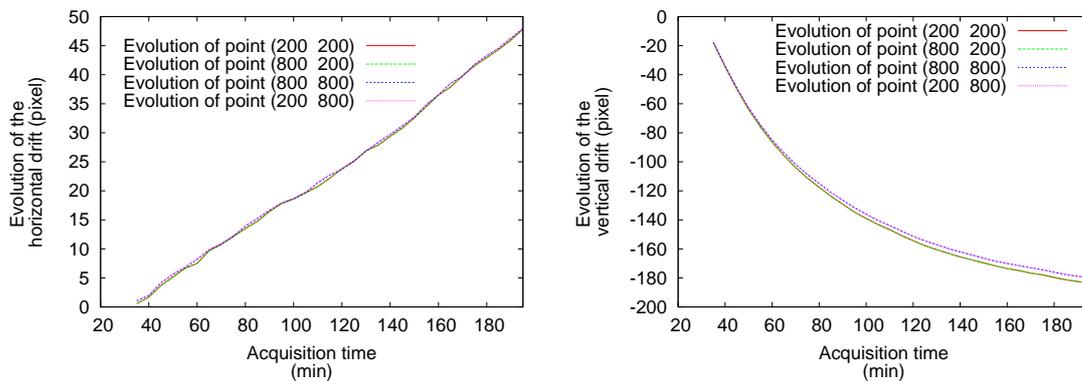


Fig. 2.15 — Evolutions of the horizontal (left) and vertical (right) drift for chosen 4 points (200 200), (800 200), (800 800) and (200 800). Drift evolutions are very similar whatever the position in the image and particularly for points located on the same row (right) because they have been acquired at near acquisition time.

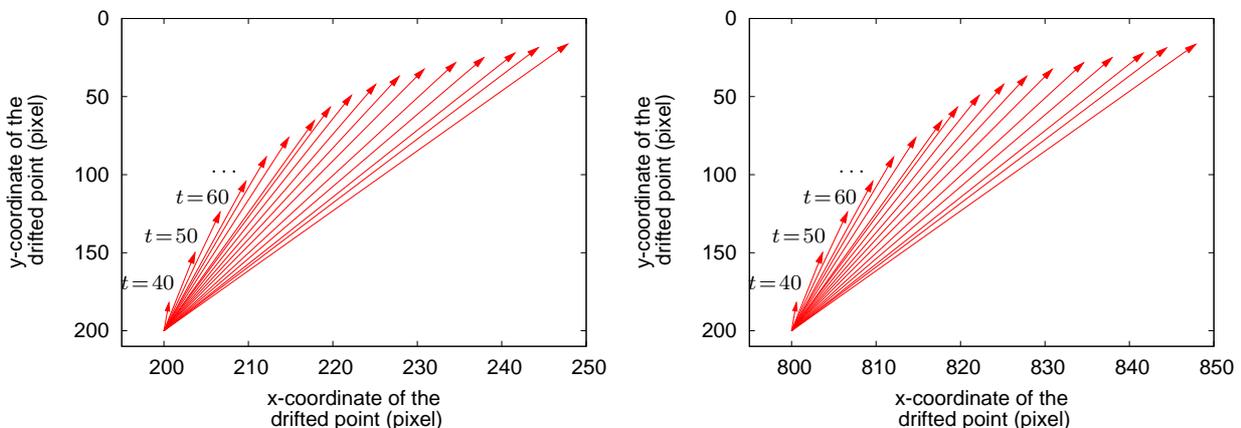


Fig. 2.16 — Evolution of the drift vector $\delta_t(t) = \begin{pmatrix} \delta_{t_x}(t) & \delta_{t_y}(t) \end{pmatrix}^\top$ every 10 minutes for the points $\mathbf{p}_1 = (200 \ 200)$ (left) and $\mathbf{p}_2 = (800 \ 200)$ (right).

The time-varying distortion function is expressed as the sum of the distorted point $\check{\mathbf{m}}$ and the drift δ_t , expressed as a function of time:

$$\check{\mathbf{m}} = \mathcal{D}_t(\check{\mathbf{m}}) = \check{\mathbf{m}} + \delta_t(t) \text{ with } \check{\mathbf{m}} = \begin{pmatrix} \check{x} \\ \check{y} \end{pmatrix} \text{ and } t = \check{x}.t_d + \check{y}.t_l \quad (2.12)$$

In order to correct SEM images for the drift, two approaches have been investigated: a) develop a global model of the drift $\delta_t(t)$ and use it locally on each image j depending on its time of acquisition t or b) develop local models $\delta_t^j(t)$, different for each image j . Note that the second approach means that the time-varying distortion will only be known locally and then, it has to be reestimated for each new acquisition, contrary to the spatially-varying distortion which is estimated once for all for a given experiment, during the calibration phase. Preliminary experiments using the first approach proved that a global model can easily be determined to represent drift evolution during time but the model will not adequately characterize the local variations in drift within an image: see Fig. 2.17. Note that the drift represented in this figure is not actually the drift but the difference between the drift at a time of acquisition t (where $20 \leq t \leq 220$ min) and the drift in the first image. However, the drift occurring during the acquisition of the first image ($20 \leq t \leq 20 + t_f = 21$ min) is negligible and can be considered as null with respect to the long period of the experiment and the magnitude of the drift.

The second approach correcting independently each image is therefore preferable. Experimentally, implementation of a local model is very simple: for each image meant to be corrected, another consecutive image of the same area and under the same operating conditions is acquired. Then, using digital image correlation⁵, the disparity map (but presented as a function of time and not as a function of position) computed between these two images and the knowledge of the time elapsed t_Δ allow to estimate the drift as a function of time.

Let's note $\check{\mathbf{m}}_i = \begin{pmatrix} \check{x}_i & \check{y}_i \end{pmatrix}^\top$ the i^{th} point in the first image and $\check{\mathbf{m}}'_i = \begin{pmatrix} \check{x}'_i & \check{y}'_i \end{pmatrix}^\top$ its correspondence in the second image. The first pixels of the first and the second image are respectively acquired at time 0 and time t_Δ . At each time t , a drift vector is associated $\delta_t(t) = \begin{pmatrix} \delta_{t_x}(t) & \delta_{t_y}(t) \end{pmatrix}^\top$.

Both points $\check{\mathbf{m}}_i$ and $\check{\mathbf{m}}'_i$ represents actually the same physical point \mathbf{M}_i on the specimen and therefore the same spatially-varying distorted projected point $\check{\mathbf{m}}_i$ at two different times:

$$\check{\mathbf{m}}_i = \check{\mathbf{m}}_i + \delta_t(t_i) \quad \text{and} \quad \check{\mathbf{m}}'_i = \check{\mathbf{m}}_i + \delta_t(t'_i)$$

⁵Ideally, in order to minimize the influence of the drift on the correlation, a DIC software working with image vectors and not image frames should be used. Because it does not exist, attention must be paid when using classic DIC software to choose a very small height of the correlation window because pixels which are neighbors vertically in the image frame are not neighbors at all in the time (ideally, the correlation window would be a row).

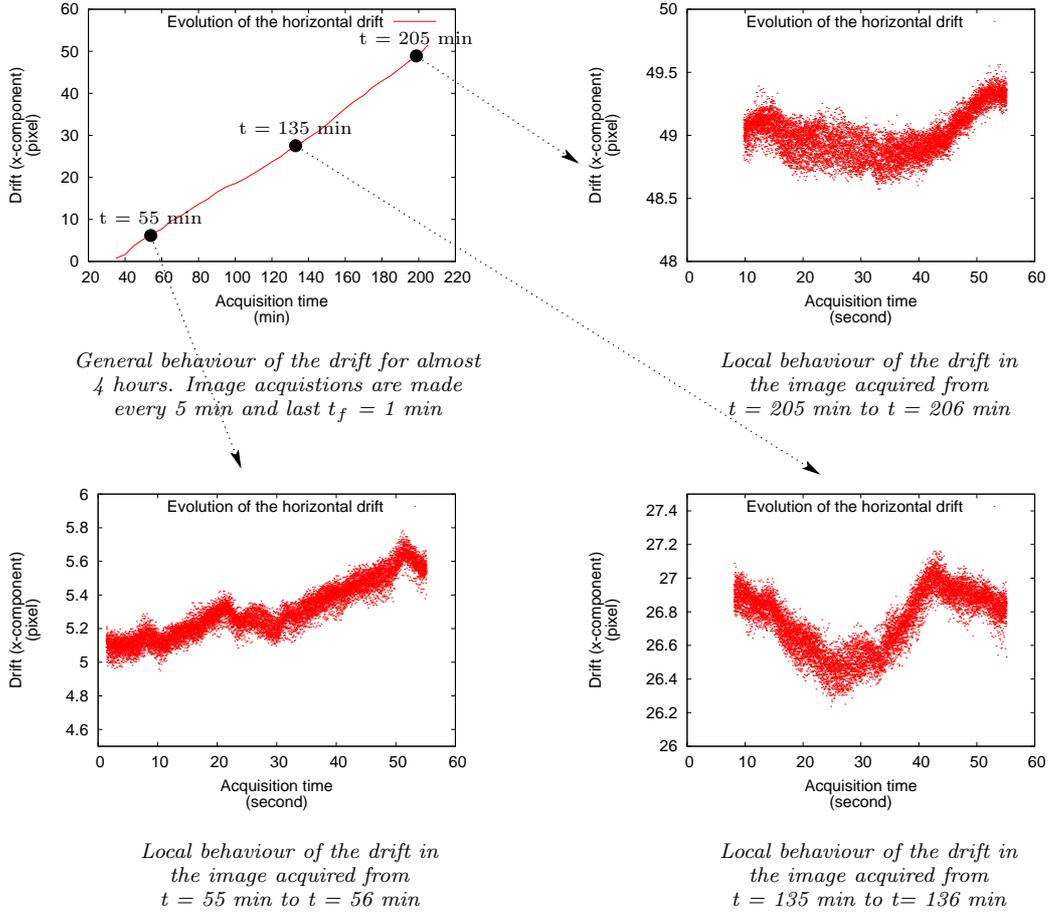


Fig. 2.17 — The general drift behavior (upper left figure) can be represented by a global model but it is not a good idea because it would not represent the important local variations of the drift in each image (upper right and bottom figures).

This leads to the following expression for the disparity:

$$\mathbf{disp}_i = \check{\check{m}}'_i - \check{\check{m}}_i = \delta_t(t'_i) - \delta_t(t_i) \quad (2.13)$$

where t_i and t'_i are respectively the times of acquisition of the pixels $\check{\check{m}}_i$ and $\check{\check{m}}'_i$:

$$t_i = \check{\check{x}}_i \cdot t_d + \check{\check{y}}_i \cdot t_l$$

and

$$t'_i = t_\Delta + \check{\check{x}}'_i \cdot t_d + \check{\check{y}}'_i \cdot t_l$$

Moreover, let's note for convenience:

$$t_{\mathbf{disp}_i} = (\check{\check{x}}'_i - \check{\check{x}}_i) t_d + (\check{\check{y}}'_i - \check{\check{y}}_i) t_l$$

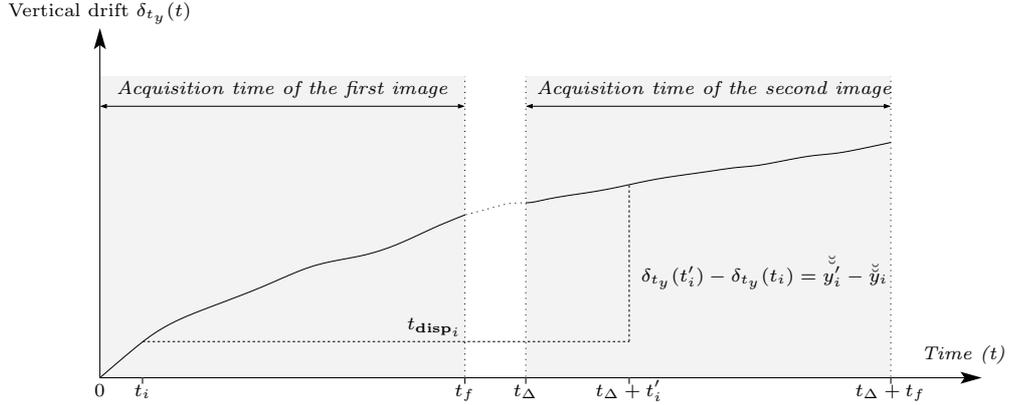


Fig. 2.18 — *Example of an unknown vertical drift function and the way to estimate it from the disparity data between two consecutive images.*

Fig. 2.18 illustrates an example of representation of the drift as a function of time with the variables defined before. Note that the delicate part of the drift function estimation is that the disparity data does not lead directly to the drift function but to drift differences (see Equ. (2.13)). Therefore, it allows to estimate the drift function only up to a constant and in order to have a unique solution, a constraint has to be added.

From the drift evolution through time (see Fig. 2.17), different models can be assumed to represent locally the time-varying distortion $\delta_t(t)$ and three have been experimented: linear, quadratic or spline curve⁶. The model equations are presented thereafter and the estimation of the parameters are discussed in Section 2.3. Note that the spline representation allows to replace the two other models by choosing a spline defined on only one knot span (see Appendix B) and by setting its degree to 1 (equivalent to linear model) or 2 (equivalent to quadratic model).

- Linear model: $\delta_t(t) = \begin{pmatrix} a_x & a_y \end{pmatrix}^\top t$

The general linear model should be $\delta_t(t) = \begin{pmatrix} a_x & a_y \end{pmatrix}^\top t + b$ but the parameter b is null. Indeed, the constraint to enforce the unique solution is that the drift is null for the first pixel $\delta_t(0) = \begin{pmatrix} 0 & 0 \end{pmatrix}^\top$ and that leads to $b = 0$.

$$\begin{aligned} \text{Equ. (2.13)} \Rightarrow \check{m}'_i - \check{m}_i &= \begin{pmatrix} a_x & a_y \end{pmatrix}^\top (t'_i - t_i) \\ &= \begin{pmatrix} a_x & a_y \end{pmatrix}^\top (t_\Delta + t_{\text{disp}_i}) \end{aligned} \quad (2.14)$$

⁶If the disparity evolution is assumed constant, it corresponds to a linear model of drift; if the disparity evolution is assumed linear, it corresponds to a quadratic model of drift; *etc.*

- Quadratic model: $\delta_t(t) = \begin{pmatrix} a_x & a_y \end{pmatrix}^\top t^2 + \begin{pmatrix} b_x & b_y \end{pmatrix}^\top t$

Similarly to the linear model, the general quadratic model should be $\delta_t(t) = \begin{pmatrix} a_x & a_y \end{pmatrix}^\top t^2 + \begin{pmatrix} b_x & b_y \end{pmatrix}^\top t + c$ but the parameter c is null due to the constraint $\delta_t(0) = \begin{pmatrix} 0 & 0 \end{pmatrix}^\top$.

$$\begin{aligned} \text{Equ. (2.13)} \Rightarrow \check{\check{m}}'_i - \check{\check{m}}_i &= \begin{pmatrix} a_x & a_y \end{pmatrix}^\top (t_i'^2 - t_i^2) + \begin{pmatrix} b_x & b_y \end{pmatrix}^\top (t_i' - t_i) \\ &= \begin{pmatrix} a_x & a_y \end{pmatrix}^\top ((t_\Delta + t_{\text{disp}_i})^2 + 2 t_i(t_\Delta + t_{\text{disp}_i})) + \\ &\quad \begin{pmatrix} b_x & b_y \end{pmatrix}^\top (t_\Delta + t_{\text{disp}_i}) \end{aligned} \quad (2.15)$$

- Spline model: $\delta_t(t, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \left(\sum_{j=0}^{n_j} \alpha_j N_x(t) \quad \sum_{k=0}^{n_k} \beta_k N_y(t) \right)^\top$

$$\begin{aligned} \text{Equ. (2.13)} \Rightarrow \check{\check{m}}'_i - \check{\check{m}}_i &= \delta_t(t_i', \boldsymbol{\alpha}, \boldsymbol{\beta}) - \delta_t(t_i, \boldsymbol{\alpha}, \boldsymbol{\beta}) \\ &= \left(\sum_{j=0}^{n_j} \alpha_j (N_x(t_i') - N_x(t_i)) \quad \sum_{k=0}^{n_k} \beta_k (N_y(t_i') - N_y(t_i)) \right)^\top \end{aligned} \quad (2.16)$$

The constraint to enforce to ensure the uniqueness of solution with the spline model is discussed in Section 2.3.

2.2.4 Complete Imaging System Model

Combining the projection model (Equ. (2.4) and Fig. 2.10) and the distortion model (Equ. (2.5) and Fig. 2.12) leads to the complete imaging system model of equation: $\check{\check{m}} = \mathcal{D}(\mathbf{KTM})$, represented by the diagram in Fig. 2.19.

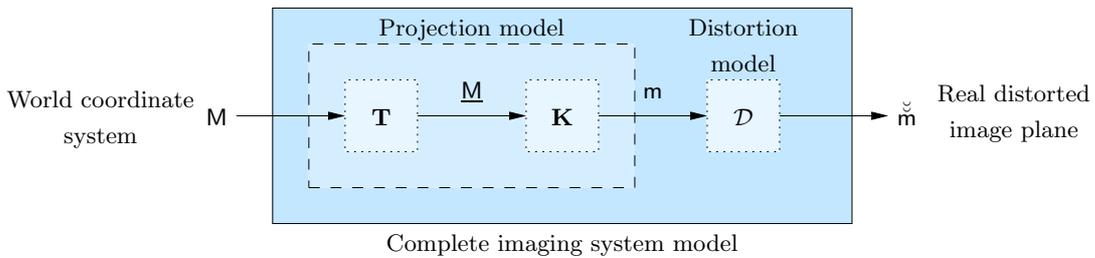


Fig. 2.19 — Schematic complete imaging system model including projection and distortion transformations.

Actually, to simplify the calibration process, the distortion function \mathcal{D} is replaced by its reciprocal $\mathcal{C} = \mathcal{D}^{-1}$, the distortion correction function (see Fig. 2.20). When the spatially-varying distortion is represented by spline functions, this approach allows to avoid issues with splines definition domain which may change during calibration process. Moreover, by this way, spline inversions are eliminated, thereby increasing the computation efficiency of the distortion removal process.

Similarly to Equ. (2.5), the distortion correction equation can be written as follows:

$$\mathbf{m} = \mathcal{C}(\check{\mathbf{m}}) = (\mathcal{C}_s \circ \mathcal{C}_t)(\check{\mathbf{m}})$$

where $\mathcal{C}(\check{\mathbf{m}}) = \mathcal{D}^{-1}(\check{\mathbf{m}})$, $\mathcal{C}_s(\check{\mathbf{m}}) = \mathcal{D}_s^{-1}(\check{\mathbf{m}})$ and $\mathcal{C}_t(\check{\mathbf{m}}) = \mathcal{D}_t^{-1}(\check{\mathbf{m}})$.

This leads to the following final model equation to calibrate (illustrated in Fig. 2.20):

$$\mathcal{C}(\check{\mathbf{m}}) = \mathbf{KTM} \tag{2.17}$$

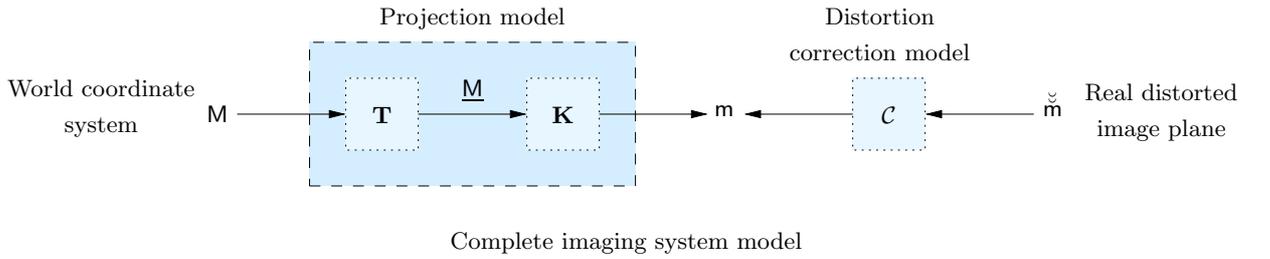


Fig. 2.20 — *Schematic imaging system model with correction distortion.*

2.3 Imaging System Calibration

Calibrating the SEM imaging system consists in determining the global transformation mapping 3D world points \mathbf{M}_i into their corresponding 2D projected points $\check{\mathbf{m}}_i$. This transformation has been modeled in the previous section and is represented by the composition of successive transformations, linear or not (see Equ. (2.17) and Fig. 2.20). The calibration of this model, or in other words, the estimation of its parameters \mathbf{T} , \mathbf{K} and \mathcal{C} requires a set of known 3D points \mathbf{M}_i of a quasi-planar calibration target associated to 2D measure points $\check{\mathbf{m}}_i^j$, projection of the i^{th} point of the calibration target in the j^{th} view. Note that because the geometry of the calibration target is not always accurately known, the 3D points \mathbf{M}_i will be reestimated as well during the calibration process [LVD98]. The experimental process to acquire the necessary data is explained in Section 2.3.2 in this chapter.

The estimation of the parameters in the least-squares sense leads to a non-linear optimization process where the sum of the magnitudes of reprojection errors⁷ ϵ_i^j is minimized:

$$\min_{\mathbf{M}_i, \mathbf{T}^j, \mathbf{K}, \mathcal{C}} \sum_i \sum_j \|\epsilon_i^j\|_2^2 \text{ where } \epsilon_i^j = \mathbf{K}\mathbf{T}^j\mathbf{M}_i - \mathcal{C}(\check{\mathbf{m}}_i^j) \quad (2.18)$$

The estimation of the 3D points \mathbf{M}_i , the extrinsic parameters of each view \mathbf{T}^j and the intrinsic parameters \mathbf{K} (corresponding to the calibration of the projective transformation) is a well-known problem referred to as bundle adjustment [TMHF00]. On the contrary, the estimation of \mathcal{C} is based on a novel approach described in the following sections. As described in section 2.2.3, measurements made using SEM images are biased by two types of distortions: a spatially-varying distortion constant for each image and a time-varying distortion different for each acquired image. This is why time-varying distortion correction functions \mathcal{C}_{t_j} are independently estimated for each view j of the calibration sequence whereas the spatially-varying distortion correction function \mathcal{C}_s is estimated using all views. Actually, for each view j , a consecutive image of the same area and under the same operating conditions is acquired (see Section 2.3.2 and Fig. 2.23) and each time-varying distortion correction function can be determined from the disparity map computed between these two images and the knowledge of the time elapsed. The non-parametric spatially-varying distortion removal approach is implemented using a relaxation algorithm that progressively improves the distortion correction function estimates using the residual reprojection errors after calibration of the projection model by bundle adjustment.

2.3.1 Calibration Procedure

The minimization problem (2.18) is solved using the following algorithm:

⁷The magnitudes of reprojection errors correspond to the distance between the estimated projected points and the actual measures.

1. Estimation of the different time-varying distortion correction functions \mathcal{C}_{t_j} (see following sections for details)
2. Correction of the original measure points $\check{\mathbf{m}}_i^j$ for drift using the time-varying distortion correction function \mathcal{C}_{t_j} corresponding to the associated view j :

$$\check{\mathbf{m}}_i^j = \mathcal{C}_{t_j}(\check{\mathbf{m}}_i^j)$$

3. Estimation of the projection model parameters and the spatially-varying distortion correction function by a relaxation technique:
 - (a) Initialization: spatially-varying distortion is considered null:

$$\mathcal{C}_s(\check{\mathbf{m}}_i^j) = \check{\mathbf{m}}_i^j$$

- (b) Spatially-varying distortion correction of the “undrifted” measure points $\check{\mathbf{m}}_i^j$:

$$\mathbf{n}_i^j = \mathcal{C}_s(\check{\mathbf{m}}_i^j)$$

- (c) Calibration of the projection model by bundle adjustment (see following sections for details):

$$\min_{\mathbf{M}_i, \mathbf{T}^j, \mathbf{K}} \sum_i \sum_j \|\epsilon_i^j\|_2^2 \text{ where } \epsilon_i^j = \mathbf{m}_i^j - \mathbf{n}_i^j$$

- (d) Estimation of the spatially-varying distortion correction function \mathcal{C}_s by fitting the data $\{\check{\mathbf{m}}_i^j, \bar{\mathbf{m}}\}$ where $\bar{\mathbf{m}} = \mathbf{n}_i^j + \omega \epsilon_i^j$ (see following sections for details)
 - (e) Repeat steps from 3b. to 3d. until convergence (the stop criterion here is relative: algorithm is stopped when the decrease of the residue value ϵ_i^j is less than 0.1%).

Note that the algorithm is presented here in the more complex case, when the spatially-varying distortion is represented by spline functions. If parametric distortion models are used, there is no relaxation procedure: steps 3a., 3b. and 3e. do not exist and steps 3c. and 3d. are gathered in a same step where distortion parameters are integrated in the bundle adjustment process.

Because effects of both distortion types occur at each acquisition, they are mixed together in the image and their separated estimation may be difficult to achieve without taking into account a part of the other distortion type. Ideally, the calibration process should estimate first 1) the time-varying distortion correction functions considering the spatially-varying one fixed, then 2) estimate the spatially-varying distortion correction

function considering the time-varying ones fixed and finally 3) iterate these two stages using a relaxation process in order to recover progressively each distortion type. This approach has been tested but in practice, the distortion types are properly separated at the first iteration and the current algorithm is then preferred for its straightforwardness.

Important steps of the algorithm are detailed in the following sections:

Step 1.: Time-Varying Distortion Correction Estimation

Based on Equ. (2.12), the time-varying distortion correction function \mathcal{C}_t can be expressed as the subtraction of the drifted point and the drift δ_t :

$$\mathcal{C}_t(\check{\mathbf{m}}) = \check{\mathbf{m}} - \delta_t(t)$$

Different models have been presented in Section 2.2.3 for δ_t and we will now explain how to calibrate them by estimating their parameters.

- Linear model: see Equ. (2.14)

t_Δ being known, the parameters a_x and a_y are easily obtained as follows:

$$\begin{pmatrix} a_x & a_y \end{pmatrix}^\top = \frac{1}{n} \sum_{i=1}^n \frac{\mathbf{disp}_i}{t_\Delta + t_{\mathbf{disp}_i}}$$

- Quadratic model: see Equ. (2.15)

With t_Δ known, the parameters of the model are estimated by a least square linear regression process:

$$\mathbf{Y}_i = \begin{pmatrix} a_x & a_y \end{pmatrix}^\top \mathbf{X}_i + \begin{pmatrix} b_x & b_y \end{pmatrix}^\top$$

with $\mathbf{Y}_i = \frac{\mathbf{disp}_i}{t_\Delta + t_{\mathbf{disp}_i}}$ and $\mathbf{X}_i = (2 t_i + t_\Delta + t_{\mathbf{disp}_i})$

- Spline model: see Equ. (2.16)

Resolving Equ. (2.16) consists in determining the parameter vectors $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ (splines coefficients) such that the distance between \mathbf{disp}_i and $\delta_t(t'_i, \boldsymbol{\alpha}, \boldsymbol{\beta}) - \delta_t(t_i, \boldsymbol{\alpha}, \boldsymbol{\beta})$ is minimal. For a least-squares optimization (assuming a normal distribution of errors), the distance to minimize is the so-called l_2 norm, defined by:

$$E_{\checkmark}(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \sqrt{\sum_{i=1}^n \|\boldsymbol{\xi}_i\|_2^2} \text{ where } \boldsymbol{\xi}_i = \mathbf{disp}_i - \delta_t(t'_i, \boldsymbol{\alpha}, \boldsymbol{\beta}) + \delta_t(t_i, \boldsymbol{\alpha}, \boldsymbol{\beta})$$

Note that minimizing E_{\checkmark} is equivalent to minimize the objective function $E = E_{\checkmark}^2$.

$$\begin{aligned} \text{Equ. (2.16)} \Rightarrow \quad E(\boldsymbol{\alpha}, \boldsymbol{\beta}) &= \begin{pmatrix} E_x(\boldsymbol{\alpha}) \\ E_y(\boldsymbol{\beta}) \end{pmatrix} \\ \text{with} \quad E_x(\boldsymbol{\alpha}) &= \sum_{i=1}^n \left\| (\check{x}'_i - \check{x}_i) - \delta_{t_x}(t'_i, \boldsymbol{\alpha}) + \delta_{t_x}(t_i, \boldsymbol{\alpha}) \right\|_2^2 \\ \text{and} \quad E_y(\boldsymbol{\beta}) &= \sum_{i=1}^n \left\| (\check{y}'_i - \check{y}_i) - \delta_{t_y}(t'_i, \boldsymbol{\beta}) + \delta_{t_y}(t_i, \boldsymbol{\beta}) \right\|_2^2 \end{aligned}$$

Equ. (2.16) does not lead to a unique solution. Indeed, if δ_t is solution, any function δ'_t defined as $\delta'_t = \delta_t + \mathbf{v}, \forall \mathbf{v} \in \mathbb{R}^2$ is also solution. To insure the uniqueness of the function to estimate, two constraints must be added to the minimization process. Because the drift is null in x and y direction at the beginning of the acquisition of the first image, we can eliminate the two degrees of freedom by fixing the constraint $\delta_t(0, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \begin{pmatrix} 0 & 0 \end{pmatrix}^\top$. For optimization problems subject to constraints, it is common to introduce the Lagrangian multipliers. They are new variables (one per constraint) gathered in a vector of vectors $\boldsymbol{\Lambda} = (\boldsymbol{\lambda}_1 \quad \boldsymbol{\lambda}_2 \quad \dots)$. Each constraint equation is multiplied by its corresponding variable, before being added to the original objective functions. For convenience, and because there is only one constraint here, λ_1 will be denoted $\boldsymbol{\lambda} = (\lambda_x \quad \lambda_y)^\top$. The new objective function to minimize becomes:

$$L(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\lambda}) = E(\boldsymbol{\alpha}, \boldsymbol{\beta}) + \boldsymbol{\lambda} \delta_t(0, \boldsymbol{\alpha}, \boldsymbol{\beta}) \quad (2.19)$$

Minimization of (2.19) is achieved by setting the derivatives equal to zero, resulting in the system of normal equations (2.20):

$$\min_{\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\Lambda}} L \Rightarrow \begin{cases} \frac{\partial L}{\partial \alpha_j} = \mathbf{0}^\top, j = 1 \dots n_j \\ \frac{\partial L}{\partial \beta_k} = \mathbf{0}^\top, k = 1 \dots n_k \\ \frac{\partial L}{\partial \boldsymbol{\lambda}} = \mathbf{0}^\top \end{cases} \Rightarrow \begin{cases} \frac{\partial E_x}{\partial \alpha_j}(\boldsymbol{\alpha}) + \lambda_x \frac{\partial \delta_{t_x}}{\partial \alpha_j}(0) = 0 \\ \frac{\partial E_y}{\partial \beta_k}(\boldsymbol{\beta}) + \lambda_y \frac{\partial \delta_{t_y}}{\partial \beta_k}(0) = 0 \\ \delta_t(0, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \mathbf{0}^\top \end{cases} \quad (2.20)$$

where:

$$\begin{aligned}
 \frac{\partial E_x}{\partial \alpha_j}(\boldsymbol{\alpha}) &= -2 \sum_{i=1}^n \left(\frac{\partial \delta_{t_x}}{\partial \alpha_j}(t_i) - \frac{\partial \delta_{t_x}}{\partial \alpha_j}(t'_i) \right) \times \left((\check{x}'_i - \check{x}_i) - \delta_{t_x}(t'_i, \boldsymbol{\alpha}) + \delta_{t_x}(t_i, \boldsymbol{\alpha}) \right) \\
 \frac{\partial E_y}{\partial \beta_k}(\boldsymbol{\beta}) &= -2 \sum_{i=1}^n \left(\frac{\partial \delta_{t_y}}{\partial \beta_k}(t_i) - \frac{\partial \delta_{t_y}}{\partial \beta_k}(t'_i) \right) \times \left((\check{y}'_i - \check{y}_i) - \delta_{t_y}(t'_i, \boldsymbol{\beta}) + \delta_{t_y}(t_i, \boldsymbol{\beta}) \right) \\
 \frac{\partial \delta_{t_x}}{\partial \alpha_j}(t) &= N_x(t), \forall t \in \mathbb{R} \\
 \frac{\partial \delta_{t_y}}{\partial \beta_k}(t) &= N_y(t), \forall t \in \mathbb{R}
 \end{aligned}$$

At last, solving for the normal equations (2.20) in order to estimate the drift function δ_t consists in solving two different linear systems: one where the unknown parameters are $\boldsymbol{\alpha}$ and λ_x and leading to the estimation of δ_{t_x} and one where the unknown parameters are $\boldsymbol{\beta}$ and λ_y and leading to the estimation of δ_{t_y} . The linear systems can simply be solved by non-iterative methods such as the pseudo-inverse method.

Step 3c.: Calibration of the projection model

Calibration of the projection model consists in estimating the parameters of the transformation \mathbf{K} (intrinsic parameters) and the parameters of the transformations \mathbf{T}^j (extrinsic parameters)⁸ corresponding to the different orientations and positions of the imaging system with respect to a coordinate system chosen as reference. This estimation problem can then be solved using the knowledge of three-dimensional points expressed in the reference coordinate system and their corresponding projected points in different images. Usually, the three-dimensional points employed are scattered on an object called calibration target. By acquiring some images of this object from different viewpoints, it is possible to simultaneously estimate the intrinsic and extrinsic parameters of the projection model and to reestimate the 3D coordinates of the supposedly known points \mathbf{M}_i of the calibration target, considering that the geometry of the calibration target is not always perfectly known. This technique called bundle adjustment was already used more than 45 years ago with photographs [Bro58] and was updated a few years ago with digital images [LVD98, TMHF00].

The estimation of the projection model parameters is achieved by minimizing the magnitudes of the reprojection errors, *i.e.* the minimization of the distances between \mathbf{m}_i^j , the

⁸The parameters of \mathbf{T}^j are usually denoted $\mathbf{T}^j = (\mathbf{R}^j, \mathbf{t}^j)$ where \mathbf{R}^j is the rotation matrix representing the orientation and \mathbf{t}^j is the translation vector representing the position of the imaging system in the view j . However, the rotation matrix is not a minimal representation of a 3D rotation since it is composed of 9 parameters for 3 rotation angles and it is therefore not appropriate for a minimization process. In order to have a minimal parameterization of the problem, rotation matrix will be converted into a minimal representation of rotation such as Euler angles or rather the rotation vector. Conversions between different rotation representations are detailed in Appendix C.

projected points in the view j of 3D points \mathbf{M}_i and \mathbf{n}_i^j , the measure-points corrected for distortion.

In the case of a non-parametric model of distortion, the measure-points \mathbf{n}_i^j are already corrected for time-varying and spatially-varying distortion at this step of the algorithm and the minimization is as follows:

$$\min_{\mathbf{M}_i, \mathbf{T}^j, \mathbf{K}} \sum_i \sum_j \|\mathbf{m}_i^j - \mathbf{n}_i^j\|_2^2 \quad \text{with } \mathbf{m}_i^j = \mathbf{K}\mathbf{T}^j\mathbf{M}_i \text{ and } \mathbf{n}_i^j = \mathcal{C}(\check{\mathbf{m}}_i^j) \quad (2.21)$$

In the case of a parametric model of distortion, the measure-points \mathbf{n}_i^j are only corrected for time-varying distortion ($\mathbf{n}_i^j = \check{\mathbf{m}}_i^j$) and the estimation of the vector of spatially-varying distortion parameters $\mathbf{d} = (k_1 \ k_2 \ k_3 \ d_1 \ d_2 \ p_1 \ p_2)$ is included in the minimization process of bundle adjustment:

$$\min_{\mathbf{M}_i, \mathbf{T}^j, \mathbf{K}, \mathbf{d}} \sum_i \sum_j \|\mathbf{m}_i^j - \mathbf{n}_i^j\|_2^2 \quad \text{with } \mathbf{m}_i^j = \mathcal{C}_s(\mathbf{K}\mathbf{T}^j\mathbf{M}_i, \mathbf{d}) \text{ and } \mathbf{n}_i^j = \mathcal{C}_t(\check{\mathbf{m}}_i^j) = \check{\mathbf{m}}_i^j \quad (2.22)$$

The minimization is carried out using a well-known iterative optimization process: the Levenberg-Marquardt technique. However, the huge number of parameters of the minimizations (2.21) or (2.22) leads to a huge over-determined equation system and solving directly the normal equations is out of question. For instance, a standard calibration experiment with a non-parametric model of distortion using 10 views and 3000 points per image leads to 60000 equations. Classical methods of matrix inversion are too computationally expensive and a very effective method using advantageously the sparse structure of matrices is implemented. This method is described in details in Appendix A.

Before the optimization process and because it requires initial guesses of each parameter to estimate, a linear calibration is first conducted. Two methods almost similar exist [RAS93, Zha98a] and allow to obtain analytically estimated values of the intrinsic and extrinsic parameters. In the case of a parametric model of distortion, the distortion parameters are assumed null before the first iteration of Levenberg-Marquardt.

To ensure the uniqueness of the minimizations (2.21) or (2.22), Garcia [Gar01] proved that 7 constraints must be enforced. Usually the 7 fixed parameters are coordinates of some points \mathbf{M}_i from the calibration target: three non-collinear 3D points are chosen to be completely or partially constrained (2 points and the z -coordinate of the third are fixed to their initial value). Note that in the case of a non-parametric model of distortion, these constraints can be omitted in the bundle adjustment to be enforced during the spline fitting (see Step 3d.). Indeed, the bundle adjustment is minimizing Equ. (2.21): $\sum_i \sum_j \|\mathbf{K}\mathbf{T}^j\mathbf{M}_i - \mathcal{C}_s(\check{\mathbf{m}}_i^j)\|_2^2$ and the uniqueness constraints can be enforced either on the projection model parameters, or on the 3D points \mathbf{M}_i , or on the spline \mathcal{C}_s .

Fig. 2.21 illustrates the whole process:

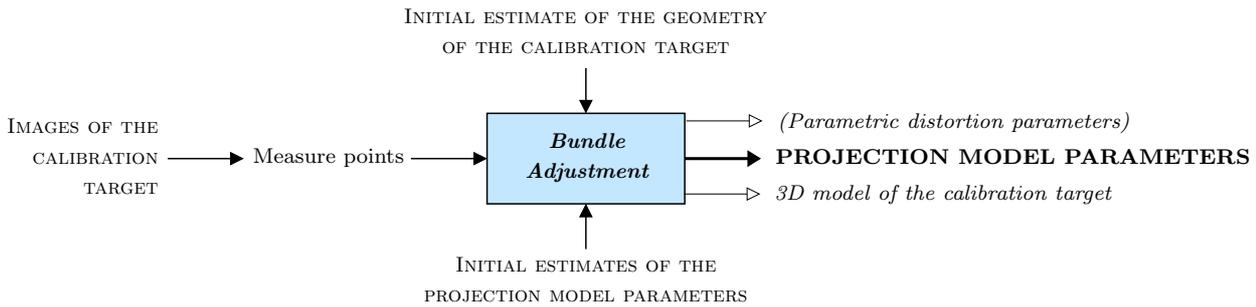


Fig. 2.21 — *In addition to the measure points (obtained from fiducial marks or speckle pattern images), the bundle adjustment approach used for the calibration needs initial guesses for the 3D shape of the calibration target and for the projection model parameters (obtained by linear calibration [RAS93, Zha98a]). The purpose of the optimization process is essentially the projection model parameters refinement (and the distortion parameters estimation in the case of a parametric distortion) but in addition, the bundle adjustment refines also the 3D shape of the calibration target.*

Step 3d.: Spatially-Varying Distortion Correction Estimation

Spatially-varying distortion can be modeled by two ways: parametric or non-parametric model. Parametric models are added to the projection model and estimated during the bundle adjustment process (see Step 3c.). To estimate the distortion in the case of a non-parametric model and obtain the spline function coefficients, Schreier *et al.* [SGS04] developed a methodology that uses images of at least two known translated motions of a plane calibration target. Even if this approach is relatively straightforward, our methodology further simplifies the process by allowing the user to employ a non-planar object and general unspecified motions. The original idea of our methodology is to consider that the remaining reprojection errors after time-varying distortion correction and calibration of the projection model correspond to the errors due to the non-correction of the spatially-varying distortion. Nevertheless, the spatially-varying distortion correction cannot be recovered directly because without the real spatially-varying distortion correction before the calibration of the projection model, a part of this distortion is included in the parameters estimated by the bundle-adjustment (particularly the reestimated 3D shape of the target). Therefore, the spatially-varying and the projection model parameters have to be estimated progressively using a relaxation process. However, the rate of convergence is very slow using directly this approach and to minimize the number of iterations (see Fig. 2.22), the principle of the S.O.R. (Successive Over Relaxation) [Kul61, Ost53] is used: the residue value ϵ_i^j is scaled by a factor ω , called S.O.R. parameter. Optimal value of this parameter can be determined *a priori* [LT87].

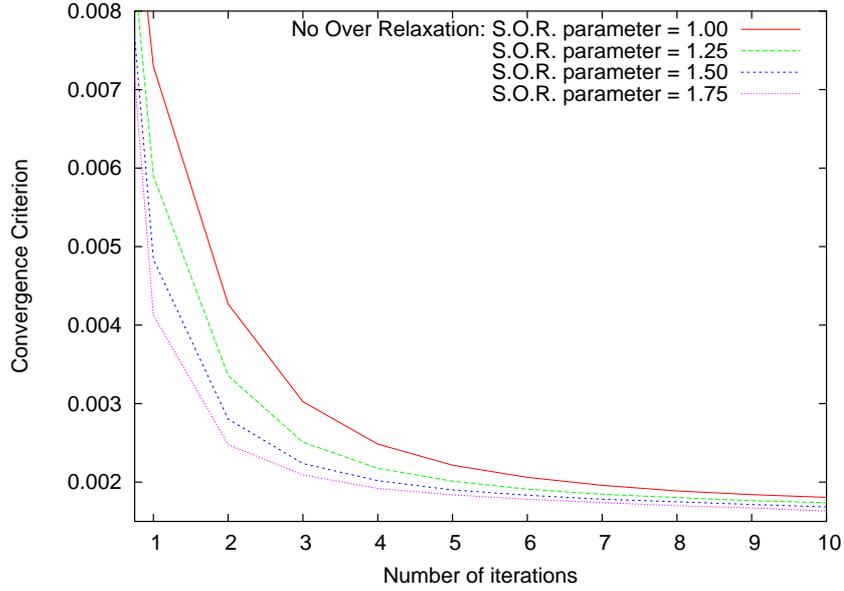


Fig. 2.22 — *Influence of the S.O.R. parameter on the rate of convergence of relaxation algorithm: we notice that the S.O.R. method is particularly efficient for the first iterations.*

The distortion correction fields in x and y direction are represented by the two 3D surfaces defined by the measure-points corrected for drift \check{m} and the coordinates $(\check{x} \ \check{y})^\top$ of $\check{m} = n_i^j + \omega \epsilon_i^j$. Let's note \mathcal{S}_x the surface composed of the n 3D points $(\check{x}_i \ \check{y}_i \ \check{x}_i)^\top, i = 1 \dots n$ and \mathcal{S}_y composed of the points $(\check{x}_i \ \check{y}_i \ \check{y}_i)^\top$. The distortion correction function is denoted $\mathcal{C}_s(\check{m}, \alpha, \beta)$ and defined as follows:

$$\mathcal{C}_s(\check{m}_i, \alpha, \beta) = \begin{pmatrix} \mathcal{C}_{s_x}(\check{m}_i, \alpha) \\ \mathcal{C}_{s_y}(\check{m}_i, \beta) \end{pmatrix} \quad \text{where} \quad \mathcal{C}_{s_x}(\check{m}_i, \alpha) = \sum_{j=0}^{n_j} \sum_{k=0}^{n_k} \alpha_{j,k} N_j(\check{x}_i) N_k(\check{y}_i)$$

$$\text{and} \quad \mathcal{C}_{s_y}(\check{m}_i, \beta) = \sum_{j=0}^{n_j} \sum_{k=0}^{n_k} \beta_{j,k} N_j(\check{x}_i) N_k(\check{y}_i)$$

It is estimated by fitting the surface \mathcal{S}_x with the spline \mathcal{C}_{s_x} and the surface \mathcal{S}_y with \mathcal{C}_{s_y} . The fitting is an approximation procedure which consists in determining the parameters vector α (respectively β) of the spline function \mathcal{C}_{s_x} (respectively \mathcal{C}_{s_y}) such that the distance between the surfaces defined by the spline and \mathcal{S}_x (respectively \mathcal{S}_y) is minimal. Similarly to previous minimizations, we assume a normal distribution of errors and use the l_2 norm to define the objective function E . For convenience, only the equations for the fitting of \mathcal{S}_x with the spline \mathcal{C}_{s_x} are given but the others can be easily obtained by replacing x by y and α by β in each expression.

$$E(\boldsymbol{\alpha}) = \sum_{i=1}^n \epsilon_i^2 \text{ where } \epsilon_i = \bar{x}_i - \mathcal{C}_{s_x}(\check{x}_i, \check{y}_i, \boldsymbol{\alpha})$$

or expressed in matrix form:

$$E(\boldsymbol{\alpha}) = \|\boldsymbol{\epsilon}\|_2^2 \text{ where } \boldsymbol{\epsilon} = \mathbf{x} - \mathbf{C}_x \boldsymbol{\alpha}$$

with:

$$\mathbf{C}_x = \begin{pmatrix} N_{0,0}^0 & \dots & N_{0,k}^0 & \dots & N_{0,n_k}^0 & N_{1,0}^0 & \dots & N_{n_j,n_k}^0 \\ \vdots & & \vdots & & \vdots & \vdots & & \vdots \\ N_{0,0}^i & \dots & N_{0,k}^i & \dots & N_{0,n_k}^i & N_{1,0}^i & \dots & N_{n_j,n_k}^i \\ \vdots & & \vdots & & \vdots & \vdots & & \vdots \\ N_{0,0}^n & \dots & N_{0,k}^n & \dots & N_{0,n_k}^n & N_{1,0}^n & \dots & N_{n_j,n_k}^n \end{pmatrix} \text{ where } N_{j,k}^i = N_j(\check{x}_i)N_k(\check{y}_i)$$

and $\mathbf{x} = (\bar{x}_0, \dots, \bar{x}_i, \dots, \bar{x}_n)^\top$

Let's note $\mathcal{C}_{s_x}^i(\boldsymbol{\alpha}) = \mathcal{C}_{s_x}(\check{x}_i, \check{y}_i, \boldsymbol{\alpha})$. The minimization is achieved by setting the derivatives equal to zero, resulting in the normal equation (2.23):

$$\begin{aligned} \min_{\boldsymbol{\alpha}} E &\Rightarrow \frac{\partial E}{\partial \boldsymbol{\alpha}} = \mathbf{0}^\top \\ &\Rightarrow 2 \sum_{i=1}^n \frac{\partial \mathcal{C}_{s_x}^i}{\partial \boldsymbol{\alpha}} (\mathcal{C}_{s_x}^i(\boldsymbol{\alpha}) - \bar{x}_i) = \mathbf{0}^\top \end{aligned} \quad (2.23)$$

$$\text{with } \frac{\partial \mathcal{C}_{s_x}^i}{\partial \boldsymbol{\alpha}} = \left(N_{0,0}^i \dots N_{0,k}^i \dots N_{0,n_k}^i \ N_{1,0}^i \dots N_{n_j,n_k}^i \right)^\top$$

In matrix form, this is expressed by the following linear system:

$$(\mathbf{C}_x^\top \mathbf{C}_x) \boldsymbol{\alpha} = \mathbf{C}_x^\top \mathbf{x} \quad (2.24)$$

Note that the size of $\mathbf{C}_x^\top \mathbf{C}_x$ and $\mathbf{C}_x^\top \mathbf{x}$ does not depend on n , the number of 3-D points. Moreover, they can be calculated directly in a very efficient manner (cumulative sum). This leads to the solution:

$$\boldsymbol{\alpha} = (\mathbf{C}_x^\top \mathbf{C}_x)^{-1} \mathbf{C}_x^\top \mathbf{x}$$

$\mathbf{C}_x^\top \mathbf{C}_x$ being a symmetric, positive definite and banded matrix, its inversion can be carried out very efficiently.

It has been shown in Step 3c. that the bundle adjustment requires some constraints to ensure the uniqueness of the estimated vector of parameters. No constraint is necessary for the spline fitting but for convenience, we can choose not to constrain the bundle

adjustment and constrain the spline fitting instead. Then, in addition to the data to fit and in order to enforce the uniqueness of the general solution, it is necessary to introduce some supplemental constraints, such as imposing the spline a given value at a given point. Consider a set of m equality constraints $\mathcal{C}_{s_x}^{n+1}(\boldsymbol{\alpha}) = c_1, \dots, \mathcal{C}_{s_x}^{n+m}(\boldsymbol{\alpha}) = c_m$. The new objective function to optimize is:

$$L_x(\boldsymbol{\alpha}, \boldsymbol{\lambda}_x) = E(\boldsymbol{\alpha}) + \lambda_{x_1}(\mathcal{C}_{s_x}^{n+1}(\boldsymbol{\alpha}) - c_1) + \dots + \lambda_{x_m}(\mathcal{C}_{s_x}^{n+m}(\boldsymbol{\alpha}) - c_m)$$

where $\lambda_{x_1}, \dots, \lambda_{x_m}$ are Lagrangian multipliers. The new normal equations are:

$$\begin{aligned} \frac{\partial L_x}{\partial \boldsymbol{\alpha}} &= \frac{\partial E}{\partial \boldsymbol{\alpha}} + \lambda_{x_1} \frac{\partial \mathcal{C}_{s_x}^{n+1}}{\partial \boldsymbol{\alpha}} + \dots + \lambda_{x_m} \frac{\partial \mathcal{C}_{s_x}^{n+m}}{\partial \boldsymbol{\alpha}} = \mathbf{0}^\top \\ \frac{\partial L_x}{\partial \boldsymbol{\lambda}} &= \left(\mathcal{C}_{s_x}^{n+1}(\boldsymbol{\alpha}) - c_1 \dots \mathcal{C}_{s_x}^{n+m}(\boldsymbol{\alpha}) - c_m \right)^\top = \mathbf{0}^\top \end{aligned}$$

which leads to the following linear system to solve:

$$\begin{pmatrix} \mathbf{C}_x^\top \mathbf{C}_x & \frac{\partial \mathcal{C}_{s_x}^{n+1}}{\partial \boldsymbol{\alpha}} & \dots & \frac{\partial \mathcal{C}_{s_x}^{n+m}}{\partial \boldsymbol{\alpha}} \\ \mathbf{C}_x^{n+1} & & & \\ \vdots & & \mathbf{0}_{m \times m} & \\ \mathbf{C}_x^{n+m} & & & \end{pmatrix} \begin{pmatrix} \boldsymbol{\alpha} \\ \lambda_{x_1} \\ \vdots \\ \lambda_{x_m} \end{pmatrix} = \begin{pmatrix} \mathbf{C}_x^\top \mathbf{x} \\ c_1 \\ \vdots \\ c_m \end{pmatrix}$$

where $\mathbf{C}_x^{n+i} = \left(N_{0,0}^{n+i} \dots N_{0,k}^{n+i} \dots N_{0,n_k}^{n+i} N_{1,0}^{n+i} \dots N_{n_j,n_k}^{n+i} \right)$

2.3.2 Experimental Process

Traditional experimental calibration process of an imaging system consists in acquiring a series of images of a planar calibration target from different viewpoints and orientations [RAS93, Zha98b, OGD97, LVD98, Gar01]. This calibration target is almost always a dedicated object (see Fig. 2.25 for an example), marked with a set of fiducial marks (*e.g.*, circles, grid intersections, *etc.*) which location has to be known more or less accurately (see [BB94, Peu94] for fiducial marks extraction). Realization of such an object at the macro-scale (for calibrating digital camera for instance) is not complex but it becomes very delicate at reduced-length scales and almost impossible for high-magnification experiments with the SEM. This is why we developed a new experimental calibration process able to use any simple object. The only requirement is that the texture of the object is random and with sufficiently small spatial wavelength so that digital image correlation can be used to locate and track small sub-regions throughout the image. Moreover, even if the method works with any calibration object of which the geometry is approximately known, it is often

chosen quasi-planar so that it is easier to give an initial guess of the calibration points M_i for the bundle adjustment. This technique has been successfully applied to several imaging systems in [CSG⁺04] and in addition to be more straightforward, calibration results show clearly that it is more accurate to use a speckle pattern textured target instead of a dedicated target with fiducial marks. Indeed, contrary to the dedicated target where the number of fiducial marks is generally small (rarely more than 100), the number of measure points with randomly-textured target is only limited by the number of possible correlation windows in the image plane, which is almost equal to the number of pixels (generally $1024 \times 764 \approx 800000$). Moreover, they can be extracted with higher accuracy than fiducial marks (the matching accuracy with digital image correlation using a speckle pattern is better than 0.01 pixels while the accuracy of the extraction of an ellipse center for example is about 0.02 pixels).

Our experimental calibration procedure differs a little compared to traditional ones because even if a series of images obtained like classic experimental calibration process should be sufficient to calibrate the projection model and the spatially-varying distortion, the time-varying distortion estimation requires another consecutive image for each acquired image (see Fig. 2.23). Thus, time-varying distortion is estimated and corrected for each pair of consecutive images while spatially-varying distortion is estimated and corrected throughout the entire image sequence. The series of images of the calibration target from different viewpoints is acquired thanks to the motorized stage of the SEM that can be translated and rotated in the plane automatically and tilted (out-of-plane rotation) manually (see Appendix D.2).

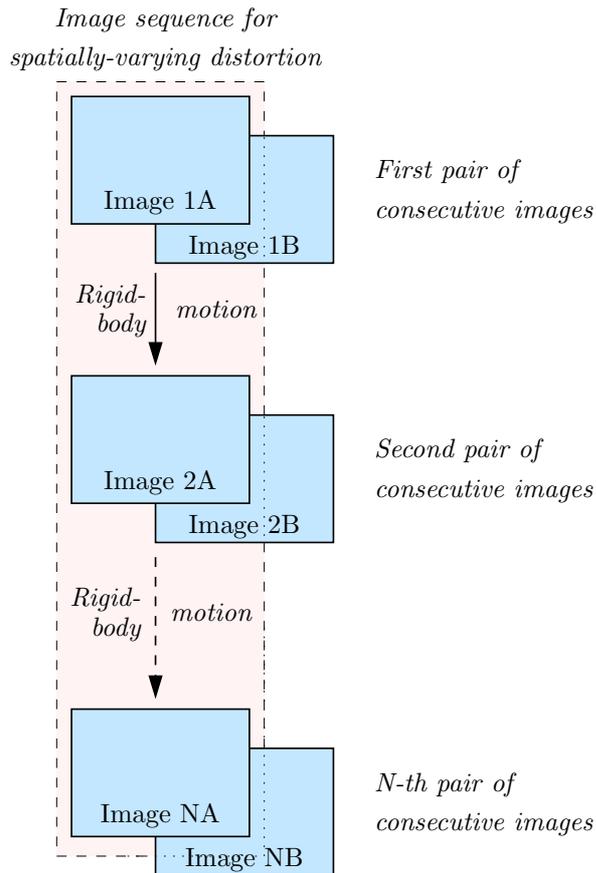


Fig. 2.23 — *Sequence of image acquisition: for each image of the series for spatially-varying distortion estimation, a consecutive one is acquired for time-varying distortion estimation. Each *A image is acquired from a different viewpoint whereas a couple of images *A and *B represents the same area from the same viewpoint.*

2.4 Experimental Results: From Macro to Nano-Scale

This section presents experimental results using different calibration methodologies including the one previously described in Section 2.3. It clearly shows the improvement in calibration accuracy of our approach with respect to other traditional methods. Thus, by eliminating the need for precision grids and employing general 3D motions of the specimen, our calibration process is both simpler and more accurate.

In order to prove the wide range of applications of our approach, experiments were conducted with different imaging systems and not only a Scanning Electron Microscope. In the case of a camera (Section 2.4.1) and a stereo-optical microscope (Section 2.4.2), there is no time-varying distortion and the calibration process is simplified. The function \mathcal{C}_t is the identity function: $\check{\mathbf{m}}_i^j = \mathcal{C}_t(\check{\mathbf{m}}_i^j) = \check{\mathbf{m}}_i^j$ and the minimization becomes:

$$\min_{\mathbf{M}_i, \mathbf{T}^j, \mathbf{K}, \mathcal{C}_s} \sum_i \sum_j \|\epsilon_i^j\|_2^2 \text{ where } \epsilon_i^j = \mathbf{K}\mathbf{T}^j\mathbf{M}_i - \mathcal{C}_s(\check{\mathbf{m}}_i^j)$$

Moreover, the calibration procedure is simplified as well and only consists of the Step 3. of the algorithm presented in Section 2.3.1.

In order to validate independently both types of distortion correction for the SEM imaging system, two experiments have been performed: the first at low magnification (Section 2.4.3) where the drift is negligible (at $200\times$ and after 3 hours, the magnitude of drift is only 25×10^{-3} pixels), the second at high magnification (Section 2.4.4) where both distortions occur.

2.4.1 Camera Imaging System Calibration

Using a QImaging Retiga 1300 CCD camera equipped with a 25 mm lens, two different sets of images have been acquired for calibration:

1. one set of 15 images of a speckle pattern target (see Fig. 2.24)

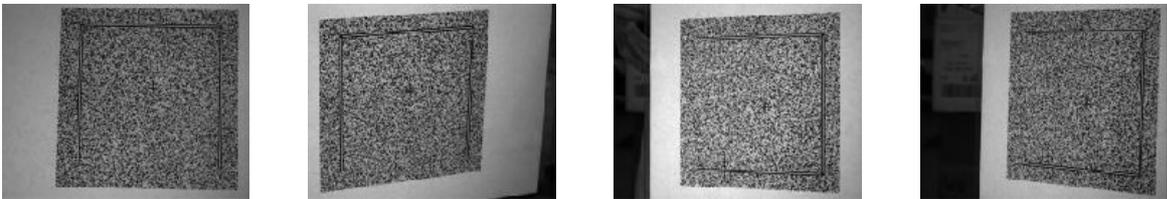


Fig. 2.24 — *Four images of the speckle Pattern calibration target (random pattern).*

2. one set of 20 images of a standard calibration target consisting of 9×9 circles (appearing as ellipses in images) (see Fig. 2.25)



Fig. 2.25 — *Four images of the standard circle calibration target.*

Using the first set of images (speckle pattern target), the simplified calibration procedure presented earlier is performed using 6500 points per image and bicubic splines of 8×8 patches⁹ for representing the distortion correction. The obtained standard deviation of the magnitudes of the reprojection errors is 0.01 pixels, with a bias (systematic error) of 0.01 pixels. If the distortion is not taken into account, the standard deviation is 0.04 pixels and the bias is 0.08 pixels and if the distortion is represented by a third-order radial distortion model, the standard deviation is 0.02 pixels. Thus, even in the presence of small amounts of distortion (see Fig. 2.26), the method is capable of a 4-fold increase in accuracy with respect to a pure projective model and a 2-fold increase compared to a parametric model of distortion.

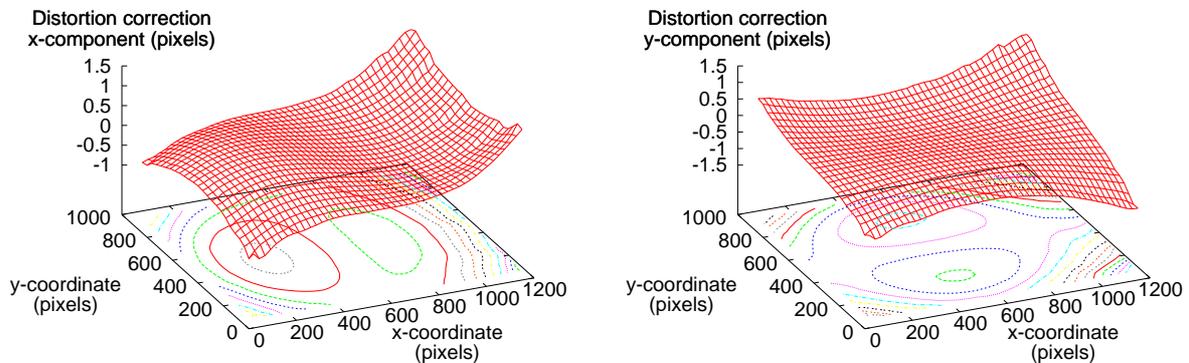


Fig. 2.26 — *The distortion correction varies from -1 pixel to 1.5 pixels horizontally (left) and from -1.5 pixels to 1.5 pixels vertically (right).*

Using the second set of images (circle target), our calibration procedure is carried out using 81 points resulting from the extraction of the ellipses centers in the 20 images. The standard deviation here is 0.03 pixels with a bias of 0.03 pixels. Note that the circle target is not adequate for estimating the distortion correction function with spline surfaces because

⁹Using 8×8 patches means that the area of interest of the image is divided in 8×8 rectangles and the data contained in each patch is approximated by a B-spline surface (see Appendix B).

81 measure points are widely insufficient for a correct spline fitting (only an average of 9 points per patch if the image is split into 3×3 patches). By comparison, considering a radial parametric model leads to a standard deviation of 0.04 pixels and a bias of 0.03 pixels whereas considering no distortion in the images leads to a standard deviation of 0.1 pixels and a bias of 0.1 pixels.

All results are gathered in Table 2.2. Note that whatever the distortion model used, the calibration results using a speckle pattern target are better than the results obtained using a classical calibration target. This is due to the difference in the number of measure-points (6500 points per image for the former and 81 points per image for the latter) and the accuracy of their extractions.

Model	Standard target	Speckle pattern target
Pinhole	0.1	0.04
Pinhole + Radial order 3	0.04	0.02
Pinhole + Spline	0.03	0.01

Table 2.2 — *Camera calibration: experimental results (standard deviation in pixels).*

2.4.2 Stereo Optical Microscope Imaging System Calibration

For the stereo optical microscope (SOM) system, the calibration process uses 20 images acquired with a Nikon SMZ-U SOM (see Fig. 2.27).

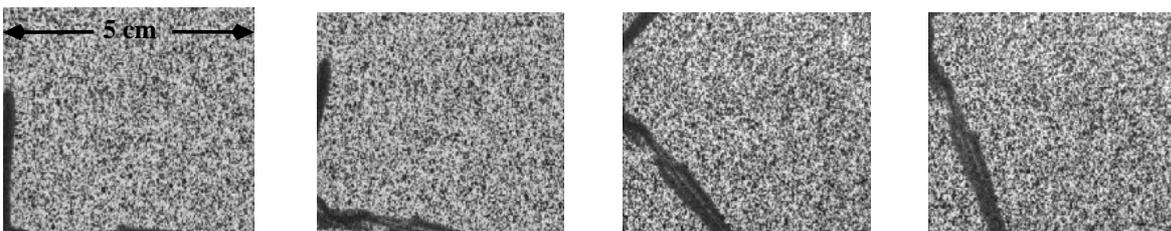


Fig. 2.27 — *Four images of the speckle pattern specimen used to calibrate the stereo-optical microscope.*

A total of 3000 points per image are used and bicubic splines with 8×8 patches are used to represent the distortion correction field. Fig. 2.28 illustrates the obtained distortion correction along x -axis and y -axis.

The standard deviation of the magnitudes of reprojection errors is 0.005 pixels with a bias of 0.01 pixels. In comparison, the standard deviation of the errors using the pinhole

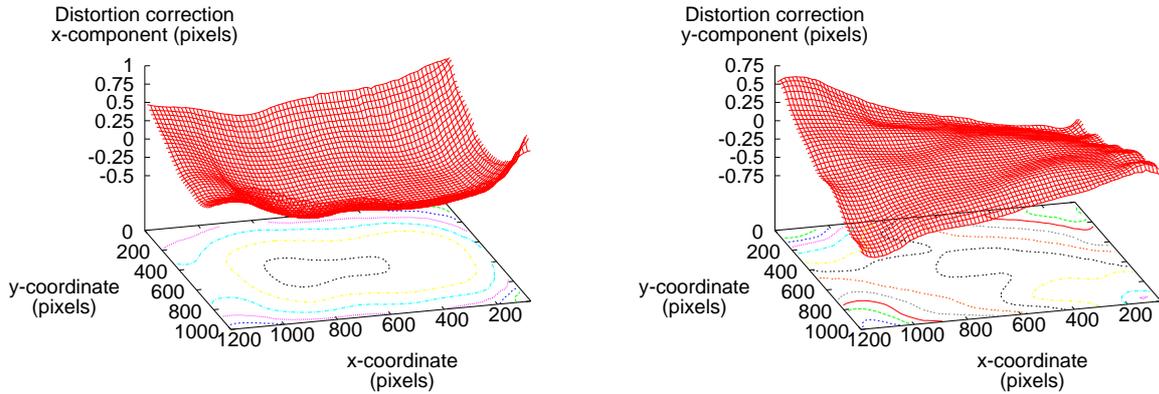


Fig. 2.28 — *The distortion correction field obtained after calibration: x -component (left), y -component (right).*

model without distortion correction is 0.08 pixels and the pinhole model with a third-order parametric radial distortion model is 0.04 pixels. Results are gathered in Table 2.3.

Model	Standard deviation (pixel)
Pinhole	0.08
Pinhole + Radial order 3	0.04
Pinhole + Spline	0.005

Table 2.3 — *Calibration of the stereo-optical microscope : experimental results.*

2.4.3 SEM Imaging System Calibration - Low Magnification

This first experiment using the SEM is conducted at low magnification and after a long waiting time (about 3 hours) in order to validate the spatially-varying distortion correction part. Indeed, the magnitude of time-varying distortion at $200\times$ and after 3 hours is about 25×10^{-3} pixels, which is negligible. The time-varying distortion correction is validated in the experiment described in Section 2.4.4.

The SEM system used in this calibration experiment is a FEI ESEM Quanta 200. A total of 24 images of a speckle-patterned calibration target ¹⁰ (see Fig. 2.29) is recorded at a $200\times$ magnification, where the target underwent different amounts of out-of-plane rotation and translation. Our calibration method is performed with 5000 points per image and bicubic splines with 8×8 patches are used to represent the distortions incurred during the imaging process.

¹⁰At low magnification, the speckle-patterned calibration target can be realized using micro-lithography (see Fig. 3.11).

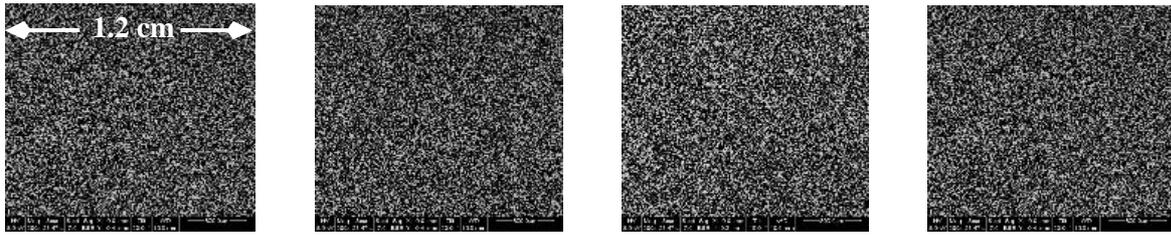


Fig. 2.29 — *A set of 24 images of a speckle-patterned calibration target is acquired with the SEM.*

Fig. 2.30 shows the distribution of the reprojection errors along y -axis. The left part of the figure demonstrates that using our calibration methodology, a non-biased normal distribution is obtained, proving that the remaining errors are essentially due to Gaussian noise. On the contrary, the right part shows that a pure projective model or a third-order radial model are far from removing correctly the distortions because the remaining errors still clearly contains a non-Gaussian error component.

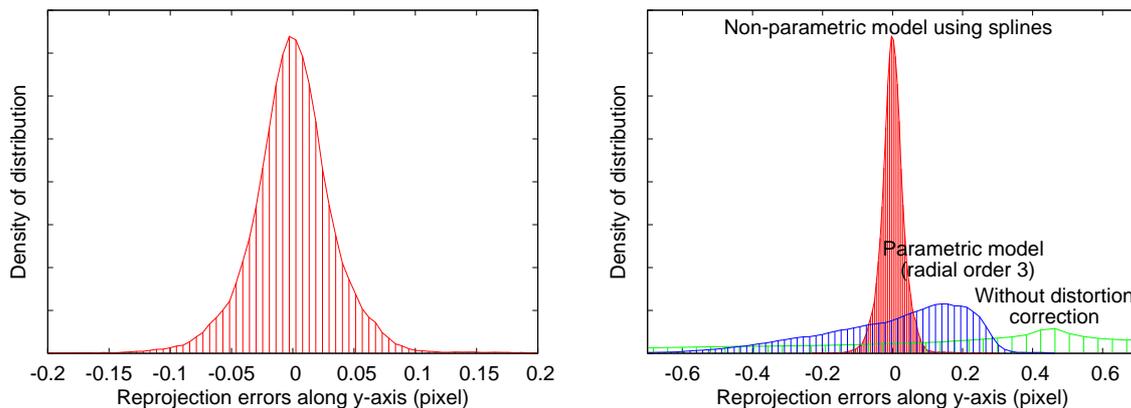


Fig. 2.30 — *The error distribution resulting from our calibration method is close to a normal distribution (left), contrary to the ones resulting from a calibration considering radial distortion or no distortion at all (right).*

The distortion correction field using spline surfaces is illustrated in Fig. 2.31. Horizontal and vertical component of the distortion correction varies respectively from -2 to 3 pixels and from -3 to 1 pixels. In the corners of the images, where the amplitude of distortion is maximum, the magnitude approaches 4 pixels.

The standard deviation of the magnitudes of reprojection errors is 0.02 pixels and the bias is 0.03 pixels using our calibration approach. By comparison, the same input data with the pinhole model and a parametric distortion correction model (radial order 3) leads to a standard deviation of 0.14 pixels and a bias of 0.24 pixels. The pinhole model without

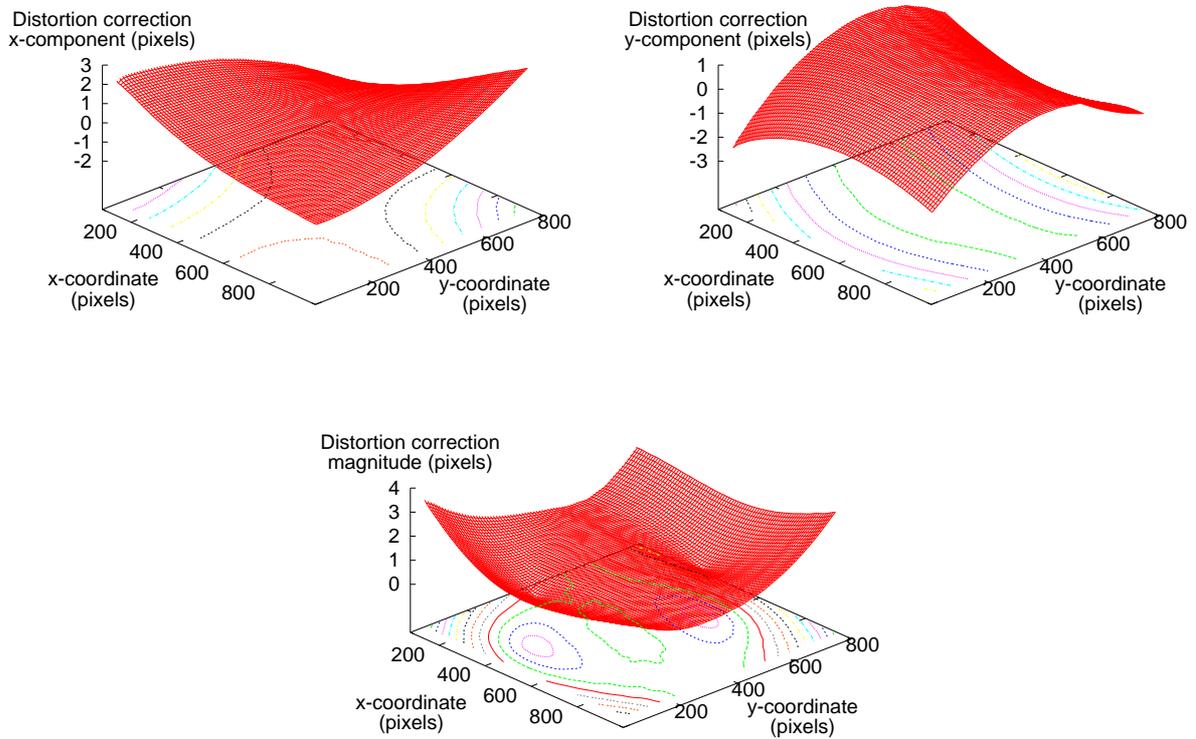


Fig. 2.31 — *Distortion correction field of the SEM at low magnification: x-component (left), y-component (right) and magnitude (bottom).*

distortion correction leads to a standard deviation of 0.43 pixels and a bias of 0.79 pixels (see Table 2.4)

Model	Standard deviation (pixel)
Pinhole	0.43
Pinhole + Radial order 3	0.14
Pinhole + Spline	0.02

Table 2.4 — *SEM calibration at low magnification: experimental results.*

This calibration experiment was the preliminary phase within an experiment aiming at the 3D reconstruction of a detail of an American penny. A calibration using directly the penny (quasi-planar object) as the calibration target has been realized as well (see Section 3.5.1 for details and experimental results).

2.4.4 SEM Imaging System Calibration - High Magnification

The objective of this second experiment using a SEM is to validate the time-varying distortion correction developed in this work. In this sense, the experiment is conducted¹¹ at high magnification (10000 \times) where the drift is quite important.

Time-varying distortion correction requires a specific experimental process described in Section 2.3.2 and in Fig. 2.23: consecutively to each image of the calibration target¹² usually acquired for the spatially-varying distortion correction, a second image of the same area and in the same operating conditions is acquired (see Fig. 2.32 and Fig. 2.33) for the time-varying distortion correction. A total of 16 images (8 pairs of consecutive images) of the calibration target are acquired using a FEI ESEM Quanta 200 and denoted 1A, 1B, 2A, . . . , 8A, 8B (each *A image is acquired from a different viewpoint whereas a couple of images *A and *B represents the same area from the same viewpoint). Fig. 2.32 presents the times t_{Δ}^j elapsed between the beginning of the acquisition of the two images of a pair j (one image acquisition lasting 95 seconds here). It presents as well the times t_{∇}^j elapsed during the j^{th} rigid-body motion (time between the end of the second image of the pair j and the first image of the pair $j + 1$).

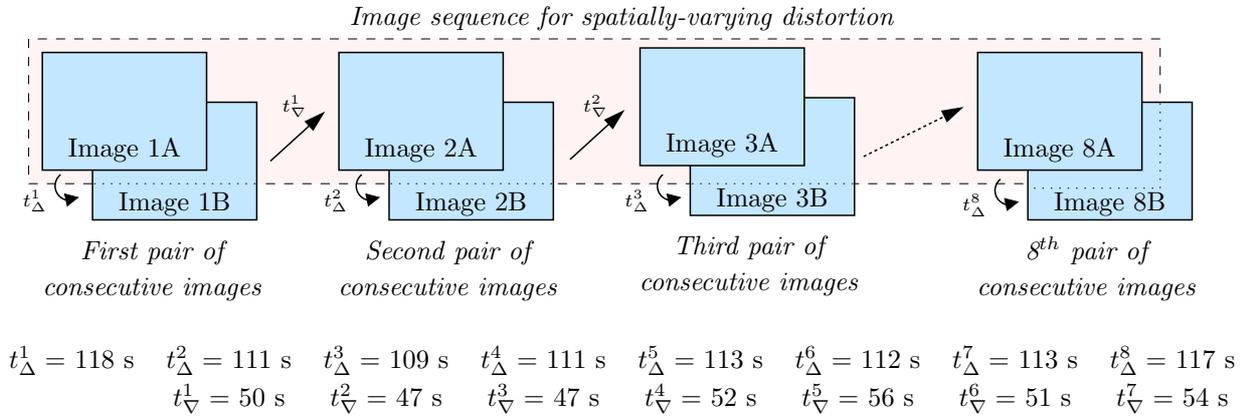


Fig. 2.32 — The time t_{Δ}^i elapsed between the acquisition of images of a pair are useful for the drift estimation. The times t_{∇}^i are given as an indication only.

The complete calibration algorithm presented in Section 2.3.1 is followed step by step:

¹¹We wish to acknowledge Dr. Ning Li from the University of South Carolina who accepted to help us by acquiring all the images of this experiment using the SEM of the Electronic Microscopy Center of USC.

¹²The calibration target used in this experiment has been realized using an original methodology developed by the Chemical Engineering Department of the University of South Carolina [CSM⁺04].

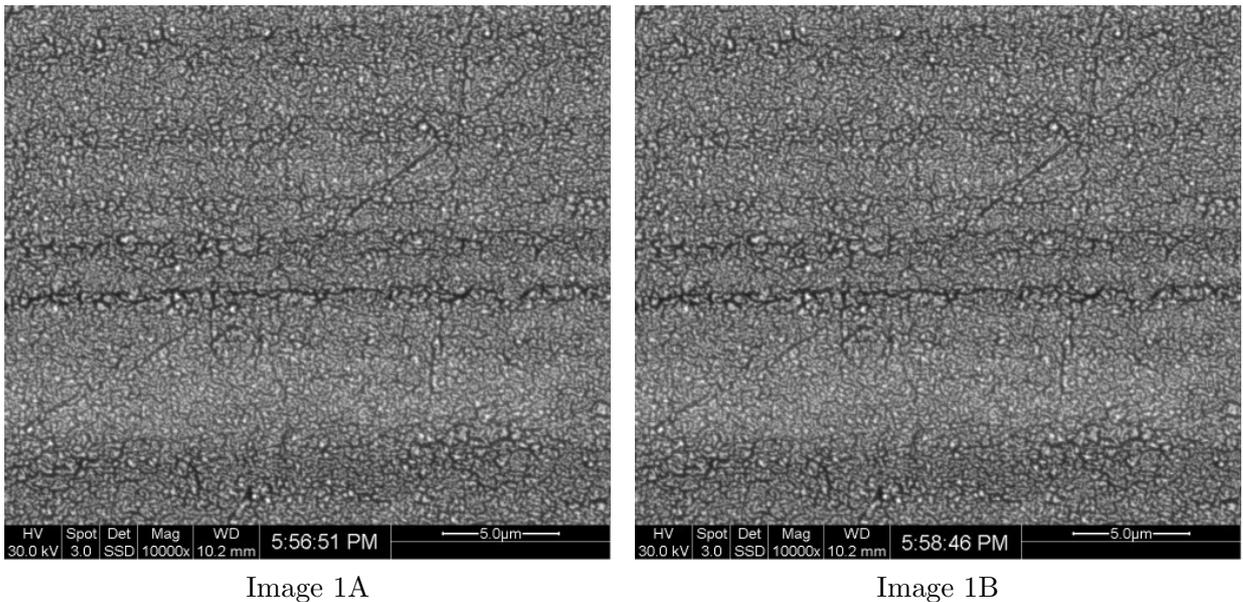


Fig. 2.33 — *Two consecutive images from the sequence of 8 pairs of images of the calibration target acquired with the SEM: images 1A and 1B represent the same area from the same viewpoint but at a different time of acquisition ($t_{\Delta}^1 = 118$ seconds).*

Step 1. and 2. of the algorithm: Time-Varying Distortion Correction

For each couple of consecutive images jA and jB , the disparity map (but presented as a function of time and not as a function of position) is computed between the two images (see Fig. 2.34). The time t_{Δ}^j elapsed between the acquisition of the 2 images is known and allows to estimate the drift function δ_t^j (see Equ. (2.13)), represented here by two spline curves of degree 5 (one for $\delta_{t_x}^j$ and one for $\delta_{t_y}^j$). Then, each image jA is corrected for the time-varying distortion.

Step 3. of the algorithm: Spatially-varying Distortion Correction

The eight images jA being now corrected for drift¹³, they are used to estimate the spatially-varying distortion while estimating the projection model parameters as well in a relaxation process (see Section 2.3.1). In this procedure, the spatially-varying distortion correction is represented by bicubic splines composed of 8×8 patches.

This new calibration methodology leads to good calibration results since the standard deviation of the magnitudes of reprojection errors is 0.03 pixels and the bias is 0.05 pixels. In order to confirm the importance of the drift correction, it is judicious to try to estimate

¹³Note that the images jB could also be corrected for drift but 8 images are already sufficient for the spatially-varying distortion estimation.

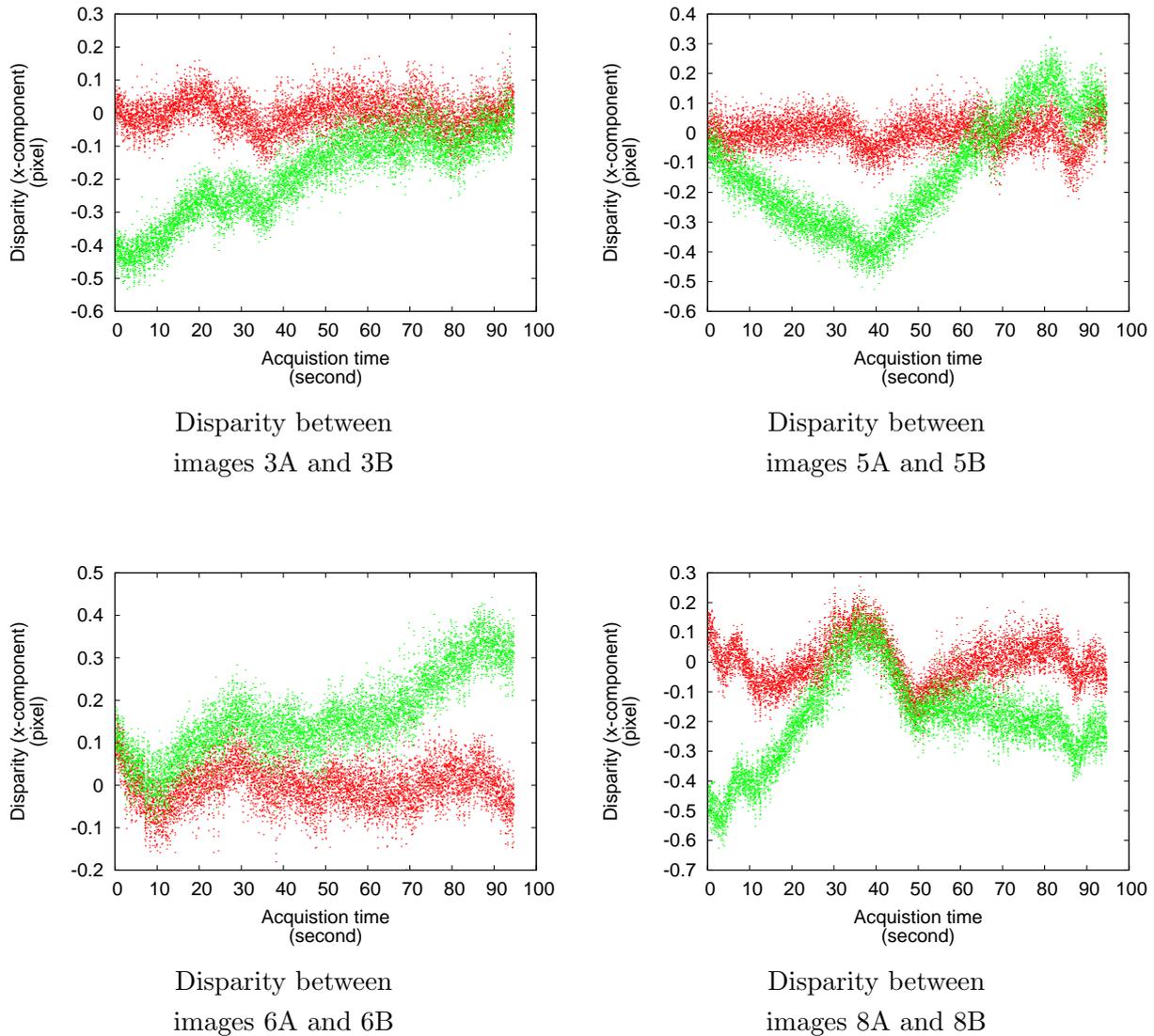


Fig. 2.34 — *Disparity computed between some pairs of consecutive images (green graphs): the disparity is supposed to be null since no motion has been undergone but due to the drift effect, it is not. After time-varying distortion correction (red graphs), the disparity vector is closer to zero but is still biased by the spatially-varying distortion (and noise).*

directly the spatially-varying distortion using the simplified calibration procedure (no preliminary time-varying distortion correction) and to compare the calibration results. In this latter case, the standard deviation of the magnitudes of reprojection errors becomes 0.1 pixels (3 times bigger) and the bias 0.2 pixels (4 times bigger). Comparison of the two distributions of reprojection errors along x -axis is illustrated in Fig. 2.35 and shows the distinct accuracy improvement when the time-varying distortion is taken into account.

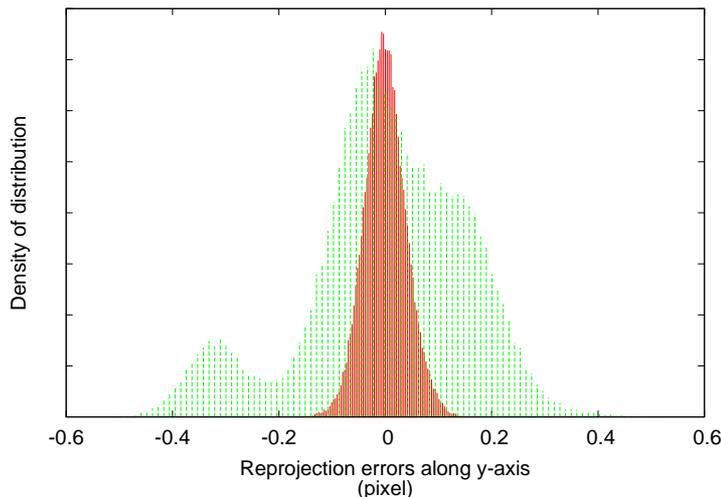


Fig. 2.35 — *Contrary to the reprojections errors obtained after the complete calibration procedure (time-varying distortion correction + spatially-varying distortion correction) represented by the red graph, the reprojections errors clearly contains a non-Gaussian error component (green graph) when the drift effect is neglected.*

Fig. 2.36 represents the reprojection errors of a view depending on the position in the image in the cases of a complete and a simplified calibration procedure. In the latter case, the reprojection errors interestingly show that a remaining deterministic component has not been corrected. This remaining part is clearly due to a time-varying distortion (dependent on the acquisition time and not on the spatial position in the image). Indeed, we can notice that it is almost constant for a given row (along x -axis) and variable for a given column (along y -axis). This is because contrary to horizontal neighbors, vertical neighbors in the image are not neighbors in the time. It proves not only that a time-varying distortion actually exists for high-magnification experiments but also that it cannot be included in the spatially-varying distortion part. The two distortion types are clearly separated and they absolutely need to be both taken into account if accurate measurements are to be made.

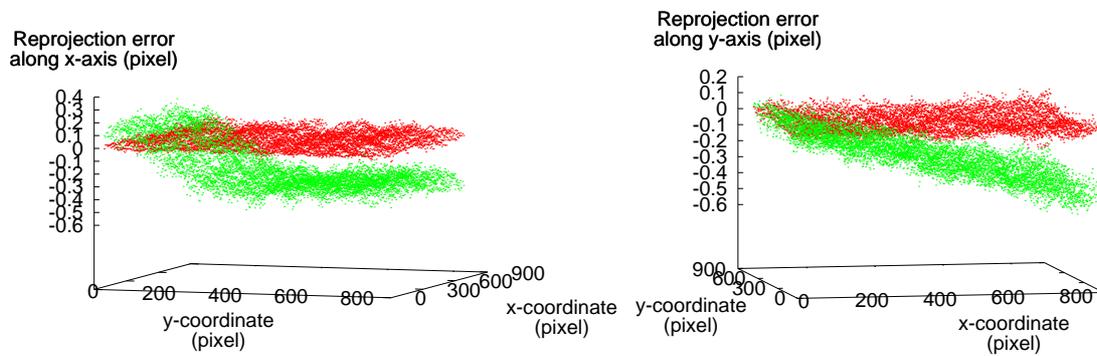


Fig. 2.36 — *The red graph represents the reprojections errors after a complete calibration procedure whereas the green graph represents them when the time-varying distortion is not preliminarily taken into account. The green graph interestingly shows that the remaining errors are due to a time-varying distortion.*

Bibliography

- [AHD⁺00] M. Agrawal, D. Harwood, R. Duraiswami, L. Davis, and P. Luther. Three Dimensional Ultrastructure from Transmission Electron Microscope Tilt Series. In *Second Indian Conference on Vision, Graphics and Image Processing (ICVGIP 2000)*, Bangalore (India), 2000.
- [BB94] R. Brand, P. Mohr and P. Bobet. Distorsions optiques: corrections dans un modèle projectif. In *Proceedings of 9th Congrès AFCET, Reconnaissances des Formes et Intelligence Artificielle (RFIA '94)*, pages 87–98, Paris (France), Jan 1994.
- [Bey92] Horst A. Beyer. Accurate Calibration of CCD-cameras. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR'92)*, pages 96–101, Urbana Champaign, (IL, USA), Jun 1992.
- [Bro58] D.C. Brown. A solution to the general problem of multiple station analytical stereotriangulation. Technical Report 23, RCA Data Reduction, Patrick Air Force base, Florida (USA), 1958.
- [Bro71] D. C. Brown. Lens Distortion for Close-Range Photogrammetry. *Photometric Engineering*, 37(8):855–866, 1971.
- [CSG⁺04] Nicolas Cornille, Hubert W. Schreier, Dorian Garcia, Michael A. Sutton, Stephen R. McNeill, and Jean-José Orteu. Gridless Calibration of Distorted Visual Sensors. In *Proceedings of Colloque Photomécanique 2004*, Albi (France), 2004.
- [CSM⁺04] S. A. Collette, M. A. Sutton, P. Miney, A. P. Reynolds, Xiaodong Li, P. E. Colavita, W. A. Scrivens, Y. Luo, T. Sudarshan, P. Muzykov, and M. L. Myrick. Development of patterns for nanoscale strain measurements: I. fabrication of imprinted au webs for polymeric materials. *Nanotechnology*, 15(12):1812–1817, December 2004.
- [DF95] F. Devernay and O. Faugeras. From Projective to Euclidean Reconstruction. Technical Report RR-2725, INRIA, Nov 1995.
- [Fau93] Olivier Faugeras. *Three-Dimensional Computer Vision: A Geometric Viewpoint*. MIT Press, 1993. ISBN 0-262-06158-9.
- [Fau95] Olivier Faugeras. Stratification of 3D Vision: Projective, Affine and Metric Representations. *Journal of the Optical Society of America A*, 12(3):465–484, Mar 1995.

- [Gar01] Dorian Garcia. *Mesure de formes et de champs de déplacements tridimensionnels par stéréo-corrélation d'images*. PhD thesis, Institut National Polytechnique de Toulouse (France), Dec 2001.
- [HA00] M. Hemmleb and J. Albertz. Microtopography - The Photogrammetric Determination of Friction Surfaces. In *IAPRS, Vol. XXXIII*, Amsterdam, 2000.
- [HAS⁺96] M. Hemmleb, J. Albertz, M. Schubert, A. Gleichmann, and J. M. Köhler. Digital microphotogrammetry with the scanning electron microscope. In *International Society for Photogrammetry and Remote Sensing*, 1996.
- [Kul61] H. E. Kulsrud. A practical technique for the determination of the optimum relaxation factor of the successive over-relaxation method. In *Communications of the ACM*, volume 4, pages 184–187, 1961.
- [LT87] P. Lascaux and R. Théodor. *Analyse numérique matricielle appliquée à l'art de l'ingénieur*. Masson, 1987. ISBN 2-90360-760-5.
- [LTCY98] A.J. Lacey, S. Thacker, S. Crossley, and R.B. Yates. A multi-stage approach to the dense estimation of disparity from stereo SEM images. *Image and Vision Computing*, 16:373–383, 1998.
- [LVD98] J-M. Lavest, M. Viala, and M. Dhome. Do we really need an accurate calibration pattern to achieve a reliable camera calibration? In *Proceedings of 5th European Conference on Computer Vision - ECCV'98*, volume 1, pages 158–174, Freiburg (Germany), Jun 1998.
- [OGD97] J.-J. Orteu, V. Garric, and M. Devy. Camera Calibration for 3-D Reconstruction: Application to the Measure of 3-D Deformations on Sheet Metal Parts. In *European Symposium on Lasers, Optics and Vision in Manufacturing*, Munich (Germany), Jun 1997.
- [Ost53] A. Ostrowski. On Over and Under Relaxation in the Theory of the Cyclic Single Step Iteration. *Mathematical Tables and Other Aids to Computation*, 7(43):152–159, 1953.
- [Peu93] Bernard Peuchot. Camera Virtual Equivalent Model – 0.01 Pixel Detector. In *Proceedings of Computerized Medical Imaging and Graphics*, volume 17, pages 289–294, Jul 1993.
- [Peu94] B. Peuchot. Utilisation de détecteurs subpixels dans la modélisation d'une caméra. In *Proceedings of 9th Congrès AFCET, Reconnaissances des Formes et Intelligence Artificielle (RFIA '94)*, pages 691–695, Paris (France), Jan 1994.

- [RAS93] O. Ravn, N. A. Andersen, and A. T. Sorensen. Auto-calibration in Automation Systems using Vision. In *3rd International Symposium on Experimental Robotics (ISER'93)*, pages 206–218, Kyoto (Japan), Oct 1993.
- [SGS04] H. Schreier, D. Garcia, and M. A. Sutton. Advances in Light Microscope Stereo Vision. *Experimental Mechanics*, 44(3):278–288, 2004.
- [SHL95] W. Shih, Y-P. Hung, and W-S. Lin. When should we consider lens distortion in camera calibration? *Pattern Recognition*, 28(3):447–461, 1995.
- [SRK⁺02] O. Sinram, M. Ritter, S. Kleindiek, A. Schertel, H. Hohenberg, and J. Albertz. Calibration of an SEM, Using a Nano Positioning Tilting Table and a Microscopic Calibration Pyramid. In *ISPRS Commission V Symposium*, pages 210–215, Corfu (Greece), 2002.
- [TMHF00] Bill Triggs, Philip F. McLauchlan, Richard I. Hartley, and Andrew W. Fitzgibbon. Bundle Adjustment – A Modern Synthesis. In Springer Verlag, editor, *Vision Algorithms: Theory and Practice*, volume 1883, pages 298–372, 2000.
- [VLBB⁺01] F. Vignon, G. Le Besnerais, D. Boivin, J.-L. Pouchou, and L. Quan. 3D reconstruction from scanning electron microscopy using stereovision and self-calibration. In *Physics in Signal and Image Processing*, Marseille (France), Jan 2001.
- [WCH92] J. Weng, P. Cohen, and M. Herniou. Camera Calibration with Distortion Models and Accuracy Evaluation. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 14, pages 965–980, 1992.
- [Zha98a] Zhengyou Zhang. A Flexible New Technique for Camera Calibration. Technical Report MSR-TR-98-71, Microsoft Research, Dec 1998.
- [Zha98b] Zhengyou Zhang. A New Multistage Approach to Motion and Structure Estimation by Gradually Enforcing Geometric Constraints. In *Asian Conference on Computer Vision (ACCV'98)*, pages 567–574, Hong Kong, Jan 1998.

Chapter 3

Automated 3D Reconstruction Process using Videogrammetry

The first objective of our work is to measure accurately the 3D shape of any object observed with a Scanning Electron Microscope. With the success of the stereo-correlation technique at the macro-scale, several authors [JJ95, HS97, LTCY98, VLBB⁺01] and commercial softwares [MeX, SAM] tried to apply this technique to the micro- and nano-scale using stereoscopic SEM images. Since the SEM provides only one imaging sensor, the stereo images cannot be acquired simultaneously and the specimen is generally tilted between the two acquisitions to simulate a stereo-rig. However, contrary to stereo-vision applications, the rigid-body transformation between the two viewpoints is not fixed and cannot be determined by acquisition of several images of a calibration target: stereo-vision is not directly applicable here and the unknown rigid-body motion will be recovered *a posteriori* from image analysis only (see Section 3.1).

The natural extension to this technique is the use of multiple views of the specimen (see Section 3.2), allowing not only a better accuracy in 3D reconstruction but also a complete view of the specimen. This technique is known for macro-scale applications as photogrammetry, videogrammetry or Structure-from-Motion method and has been recently applied to SEM 3D shape measurement [CGS⁺03, CGS⁺04, LBVP⁺04]. The important steps for the accuracy of the 3D reconstruction are the calibration, the distortion correction and the quality of the motion estimation. Previous works about 3D shape measurement using a SEM rarely take into account distortion correction and always neglect the drift problem. Moreover, the few authors considering distortion always use common parametric models, which are not really adapted to complex imaging systems such as the SEM [CSG⁺04]. Concerning the rigid-body motion estimation, the approach presented here consists not only in recovering them using the epipolar geometry but also in reestimating them in a global optimization process.

In addition to the accuracy, the second point in our 3D reconstruction process is its

automation. With this objective, Section 3.3 introduces interest point detectors which extract automatically feature points in the acquired images. These particular points are very interesting because they are characteristic of an image and can be robustly extracted in an image sequence. They can be matched automatically and allow then to estimate the epipolar geometry without any intervention of the user.

Section 3.4 is the last step of the reconstruction process. It presents triangulation equations for 2 views and more and gives different methods to solve them. Throughout this chapter, the 3D reconstruction will always be considered first in the general case of a perspective projection model of imaging sensor. The case of parallel projection is not more difficult but differs slightly in some steps which will be generally briefly described at the end of the sections.

Finally, Section 3.5 presents applications of videogrammetry at different length-scales: the 3D reconstruction of a penny detail at micro-scale and the 3D reconstruction of a mold at macro-scale. This last experiment is also the occasion to assess the accuracy of the videogrammetry technique by comparing with the 3D shape measurement using a laser scanning 3D digitizer.

Before entering in details of each stage and in order to facilitate the comprehension of the complete algorithm, Fig. 3.1 recapitulates the different stages of the 3D reconstruction process for a multi-view geometry and the reader may refer to it when necessary.

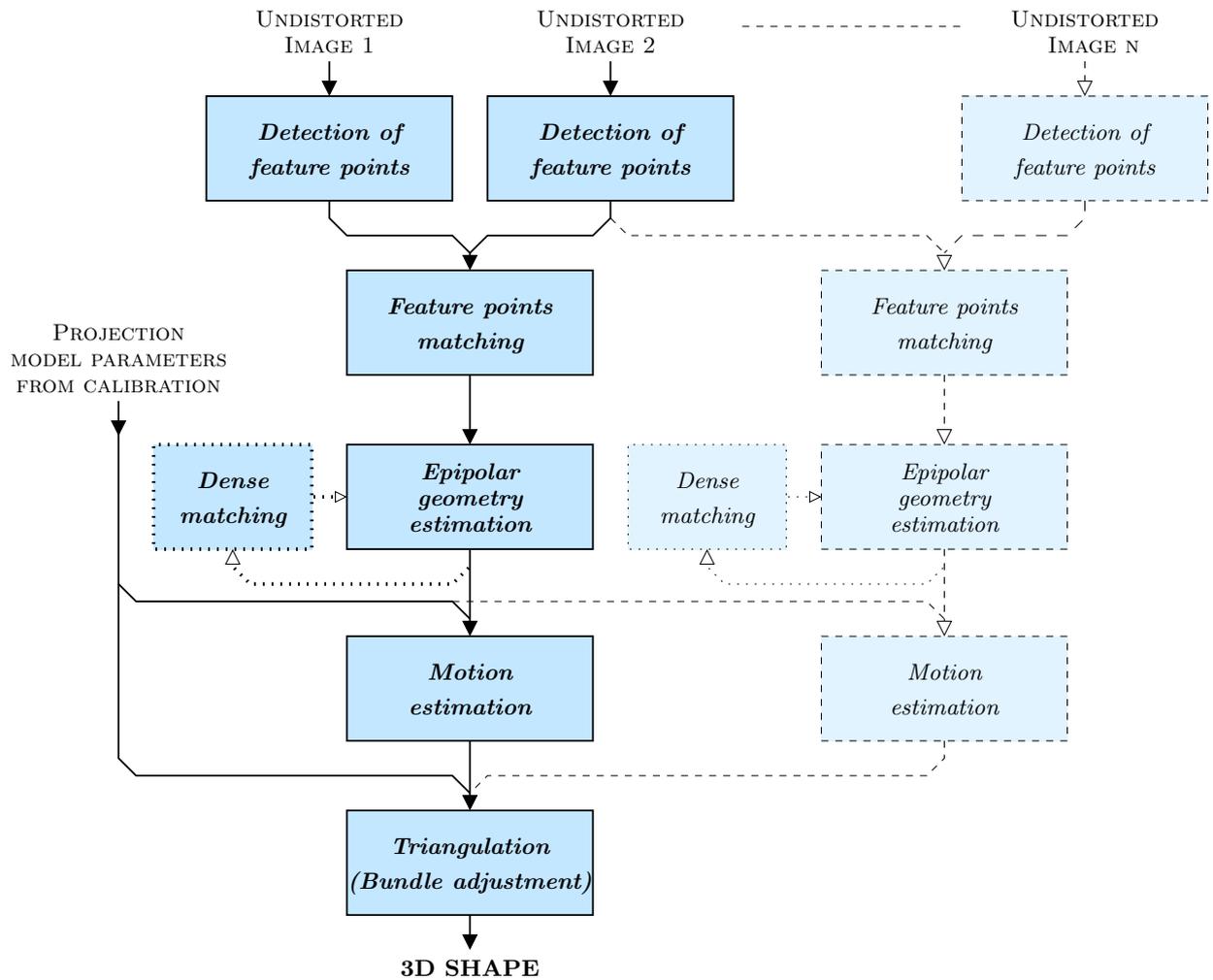


Fig. 3.1 — *Multistage process of the automated 3D shape measurement using videogrammetry.*

3.1 2-View Geometry

Before presenting in Section 3.2 the general case of multiple view geometry, we consider in this section the minimal case of multiple view geometry able to lead to 3D reconstruction: the two-view geometry. An imaging sensor is defined by a projective transformation which transforms a 3D point into a 2D point but this transformation removes the third dimension and is therefore irreversible: for a given 2D point, we do not have a unique possibility for its corresponding 3D point but an infinity along a line called projection ray (see Fig. 3.2, left). By using two imaging sensors or in our case, by moving the object and acquiring a new image from a different viewpoint, it is possible to recover the three-dimensional position of the point by triangulation (see Section 3.4). It exists indeed only one point in 3D space which corresponds to a pair of projected points (see Fig. 3.2, right).

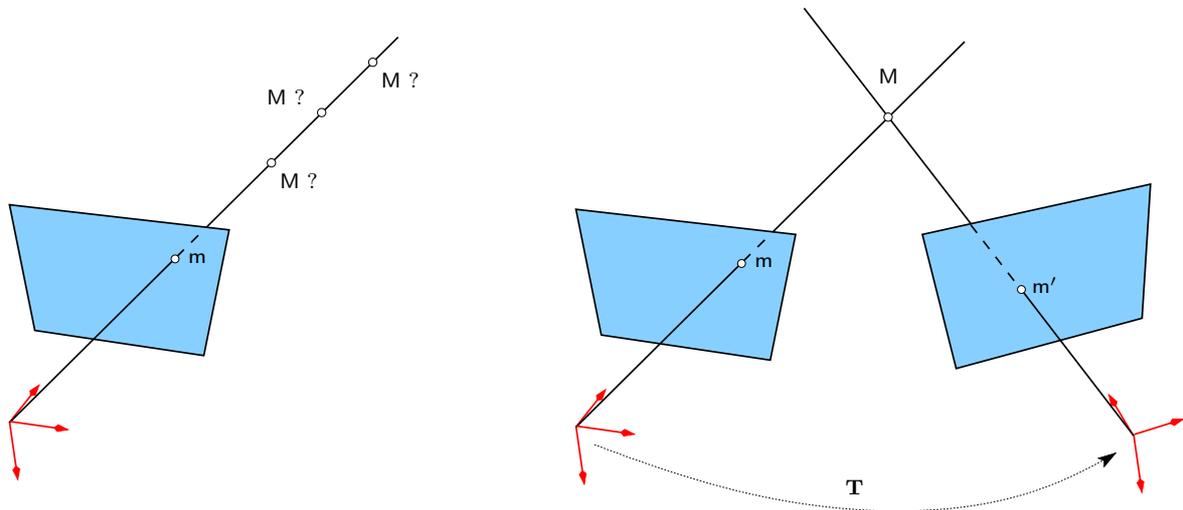


Fig. 3.2 — *One-view geometry (left): the 3D point M is known to be along the projection ray but cannot be determined; Two-view geometry (right): the 3D point is unique, it is the intersection of the two projection rays.*

Triangulation consists in determining the intersection in space of the two projections rays. Therefore, it is necessary to express these two lines in the same coordinate system, *i.e.* to know the rigid-body motion $\mathbf{T} = (\mathbf{R}, \mathbf{t})$ (\mathbf{R} is the rotation matrix and \mathbf{t} the translation vector) existing between the two viewpoints. The precision of this information being critical for the quality of the 3D reconstruction, it is preferable to recover this motion *a posteriori* from the images rather than to trust the position sensors of the SEM stage. This is possible using a geometric constraint existing between any pair of images: the epipolar geometry [Fau93, LF96, Zha96a].

The three first sections 3.1.1, 3.1.2 and 3.1.3 present the epipolar geometry, the fundamental matrix and the motion recovery in the most complex case of a general perspective

projection (low magnification). The parallel projection leads to certain simplifications described quickly in the last section.

3.1.1 Epipolar Geometry

Epipolar geometry is the relationship between correspondences of any pair of images. For a given projected point \mathbf{m} in one image, we know that its corresponding 3D point \mathbf{M} in space lies on a line called projection ray. Let's note \mathbf{l}'_m the projected line of this projection ray in a second image. Then, \mathbf{M} being on the projection ray, its projected point \mathbf{m}' in the second image have to be on the line \mathbf{l}'_m , called epipolar line of \mathbf{m} (see Fig. 3.3).

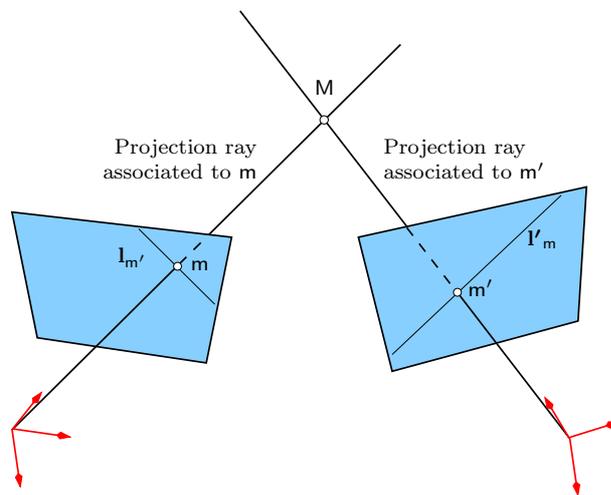


Fig. 3.3 — *Epipolar geometry: epipolar line \mathbf{l}'_m (respectively \mathbf{l}_m) is the image of the projection ray associated to \mathbf{m} (respectively \mathbf{m}').*

This is called the epipolar constraint: for a given point \mathbf{m} in one image, its corresponding point \mathbf{m}' in the second image must lie on its epipolar line \mathbf{l}'_m . Respectively, \mathbf{m} lies on \mathbf{l}_m , the epipolar line associated to \mathbf{m}' . It is algebraically expressed as follows:

$$\begin{aligned} \mathbf{m}' \text{ constrained to be on } \mathbf{l}'_m &\Rightarrow \tilde{\mathbf{m}}'^{\top} \mathbf{l}'_m = 0 \\ \text{and } \mathbf{m} \text{ constrained to be on } \mathbf{l}_m &\Rightarrow \tilde{\mathbf{m}}^{\top} \mathbf{l}_m = 0 \end{aligned}$$

Fundamental Matrix

In order to have a direct algebraic relationship between a point $\tilde{\mathbf{m}}$ in the first image and its correspondence $\tilde{\mathbf{m}}'$ in the second image, the fundamental matrix \mathbf{F} is introduced. It is a 3×3 matrix defined such that $\mathbf{F} \tilde{\mathbf{m}} = \mathbf{l}'_m$. Likewise, the fundamental matrix \mathbf{F}' represents the relationship between a point \mathbf{m}' in the second image and its correspondence \mathbf{m} in the

first image and is defined such that $\mathbf{F}' \tilde{\mathbf{m}}' = \mathbf{l}_{m'}$. Substituting \mathbf{I}'_m and $\mathbf{l}_{m'}$ in the epipolar constraints leads to:

$$\tilde{\mathbf{m}}'^{\top} \mathbf{F} \tilde{\mathbf{m}} = 0 \quad (3.1)$$

$$\text{and } \tilde{\mathbf{m}}^{\top} \mathbf{F}' \tilde{\mathbf{m}}' = 0 \quad (3.2)$$

Note that if we transpose Equ. (3.1), we obtain an equation similar to Equ. (3.2) and can deduce a relationship between the both fundamental matrices:

$$\begin{aligned} (3.1)^{\top} &\Rightarrow \tilde{\mathbf{m}}^{\top} \mathbf{F}^{\top} \tilde{\mathbf{m}}' = 0 \\ &\Rightarrow \mathbf{F}^{\top} \cong \mathbf{F}' \end{aligned}$$

where \cong is the projective equality (equality up to a scale factor): see Section 2.2.1 for details.

Essential Matrix

Equivalent to the fundamental matrix but in the imaging sensor coordinate system, the essential matrix \mathbf{E} represents the relationship between points in the first viewpoint and the second viewpoint. The unknown rigid-body motion between the two viewpoints will be extracted and recovered from this matrix. The essential matrix was introduced by Longuet-Higgins before the fundamental matrix, more than 20 years ago in [LH81]. It can be defined by two different ways:

- as a function of \mathbf{F} and the intrinsic parameters of the imaging sensor in the two positions:

$$\mathbf{E} = \mathbf{A}'^{\top} \mathbf{F} \mathbf{A} \quad (3.3)$$

where \mathbf{A} is the affine transformation of the projection model associated to the imaging sensor in the first viewpoint and \mathbf{A}' the one corresponding to the second viewpoint. Note that in our case, we have only one imaging sensor and the intrinsic parameters does not change between the two viewpoints so $\mathbf{A}' = \mathbf{A}$.

- as a function of the rigid-body transformation (\mathbf{R}, \mathbf{t}) between the two viewpoints:

$$\begin{aligned} \mathbf{E} &= \mathbf{t} \times \mathbf{R} \\ &= [\mathbf{t}]_{\times} \mathbf{R} \end{aligned} \quad (3.4)$$

where \times represents the cross product and $[\mathbf{t}]_{\times}$ is the antisymmetric matrix defined by \mathbf{t} :

$$[\mathbf{t}]_{\times} = \begin{pmatrix} 0 & -t_z & t_y \\ t_z & 0 & -t_x \\ -t_y & t_x & 0 \end{pmatrix}$$

This matrix representation of the vector \mathbf{t} allows to do a simple multiplication of matrices instead of a cross product.

The essential matrix has only five degrees of freedom: 3 for the rotation plus 3 for the translation but minus 1 because it is defined up to a scale factor.

3.1.2 Fundamental Matrix Estimation

Determining epipolar geometry in order to recover the motion between two viewpoints consists in estimating the fundamental matrix parameters from a set of point matches. The technique usually used consists in writing the linear equation corresponding to the epipolar constraint (see Equ. (3.1)), like in [Zha96b]:

Consider a point $\tilde{\mathbf{m}}_i = (x_i \ y_i \ 1)^\top$ in the first image, its corresponding point $\tilde{\mathbf{m}}'_i = (x'_i \ y'_i \ 1)^\top$ in the second image and $\mathbf{F} = \begin{pmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{pmatrix}$. Then:

$$\begin{aligned} \text{Equ. (3.1)} \quad &\Rightarrow x_i x'_i f_{11} + y_i x'_i f_{12} + x'_i f_{13} + x_i y'_i f_{21} + y_i y'_i f_{22} + y'_i f_{23} + x_i f_{31} + y_i f_{32} + f_{33} = 0 \\ &\Leftrightarrow \mathbf{u}_i \mathbf{f}^\top = 0 \end{aligned}$$

where:

$$\begin{aligned} \mathbf{u}_i &= (x_i x'_i \ y_i x'_i \ x'_i \ x_i y'_i \ y_i y'_i \ y'_i \ x_i \ y_i \ 1) \\ \text{and } \mathbf{f} &= (f_{11} \ f_{12} \ f_{13} \ f_{21} \ f_{22} \ f_{23} \ f_{31} \ f_{32} \ f_{33}) \end{aligned}$$

From n point matches, the following linear system can then be established:

$$\mathbf{U} \mathbf{f}^\top = 0 \quad \text{with } \mathbf{U} = \begin{pmatrix} \mathbf{u}_1 \\ \vdots \\ \mathbf{u}_n \end{pmatrix}$$

As the fundamental matrix \mathbf{F} is a 3×3 matrix, nine parameters should be evaluated and then nine points matches should be necessary but in fact, this number is smaller due to two particularities of \mathbf{F} :

$$\begin{aligned}
 1. \quad & \text{Equ. (3.1)} \Leftrightarrow \forall \lambda, \tilde{\mathbf{m}}'^{\top} (\lambda \mathbf{F}) \tilde{\mathbf{m}} = 0 \\
 & \Rightarrow \mathbf{F} \text{ is defined up to a scale factor}
 \end{aligned} \tag{3.5}$$

$$\begin{aligned}
 2. \quad & \left\{ \begin{array}{l} \text{Equ. (3.3) and Equ. (3.4)} \Rightarrow \mathbf{F} = \mathbf{A}'^{-\top} [\mathbf{t}]_{\times} \mathbf{R} \mathbf{A}^{-1} \\ [\mathbf{t}]_{\times} \text{ antisymmetric matrix} \Rightarrow |[\mathbf{t}]_{\times}| = 0 \end{array} \right. \\
 & \Rightarrow |\mathbf{F}| = 0
 \end{aligned} \tag{3.6}$$

Thus, only seven point matches are needed to estimate the seven independent parameters of \mathbf{F} with an appropriate parameterization. Actually, in practice, we have always more than seven point matches and we preferred to implement the *8-point algorithm* [LH81] which uses eight or more point matches.

However, because the fundamental matrix estimation is very sensitive to noise, Hartley's method has to be implemented as well [Har95]. This method is proved to lead to a better stability of the results thanks to an improvement in the condition number of the linear system. It consists in applying, independently on each image, a 2-step transformation called isotropic scaling to the coordinates before carrying out the *8-point algorithm*:

1. each point \mathbf{m}_i is translated so that the centroid of all points is at the origin:

$$\tilde{\mathbf{m}}_i = \mathbf{m}_i - \frac{1}{n} \sum_{i=1}^n \mathbf{m}_i$$

2. the coordinates of each point are scaled so that the average distance to the origin is equal to $\sqrt{2}$:

$$\hat{\mathbf{m}}_i = \frac{\tilde{\mathbf{m}}_i \sqrt{2}}{\frac{1}{n} \sum_{i=1}^n \|\tilde{\mathbf{m}}_i\|}$$

Then, the *8-point algorithm* can be performed but the obtained matrix $\hat{\mathbf{F}}$ corresponds to modified inputs $\hat{\mathbf{m}}_i$ and $\hat{\mathbf{m}}'_i$. Thus, it has to be transformed as well in order to recover the original fundamental matrix \mathbf{F} .

Let \mathbf{T} be the isotropic scaling transformation applied to each \mathbf{m}_i and \mathbf{T}' , the one applied to \mathbf{m}'_i :

$$\tilde{\hat{\mathbf{m}}}_i = \mathbf{T} \tilde{\mathbf{m}}_i \quad \text{and} \quad \tilde{\hat{\mathbf{m}}}'_i = \mathbf{T}' \tilde{\mathbf{m}}'_i$$

Then:

$$\begin{aligned}
\text{Equ. (3.1)} &\Rightarrow (\mathbf{T}'^{-1}\tilde{\mathbf{m}}'_i)^\top \mathbf{F} (\mathbf{T}^{-1}\tilde{\mathbf{m}}_i) = 0 \\
&\Rightarrow \tilde{\mathbf{m}}'^{\top}_i \underbrace{(\mathbf{T}'^{-\top} \mathbf{F} \mathbf{T}^{-1})}_{\hat{\mathbf{F}}} \tilde{\mathbf{m}}_i = 0
\end{aligned}$$

And eventually, we recover \mathbf{F} :

$$\mathbf{F} = \mathbf{T}'^\top \hat{\mathbf{F}} \mathbf{T}$$

However, the estimated fundamental matrix does not satisfy the rank-2 constraint and Tsai and Huang [TH84] suggest to enforce this singularity constraint *a posteriori* by replacing \mathbf{F} by $\hat{\mathbf{F}}$ which minimizes the Frobenius norm of $\|\mathbf{F} - \hat{\mathbf{F}}\|$ and subject to the constraint $|\hat{\mathbf{F}}| = 0$. This is done easily using a Singular Value Decomposition of \mathbf{F} .

Eventually, this linear method is still not accurate enough and its result is only used as initial guess for an optimization approach. Indeed, the non-linear methods are better because they minimize a physically meaningful quantity like distance of points to epipolar lines or distances between observation and reprojection. We chose to implement the former with the adapted fundamental matrix parameterization [LF96].

The criterion to minimize is:

$$\sum_{i=1}^n d^2(\tilde{\mathbf{m}}'_i, \mathbf{F}\tilde{\mathbf{m}}_i) + d^2(\tilde{\mathbf{m}}_i, \mathbf{F}^\top \tilde{\mathbf{m}}'_i)$$

where $d(\tilde{\mathbf{m}}'_i, \mathbf{F}\tilde{\mathbf{m}}_i)$ is the Euclidean distance of point $\tilde{\mathbf{m}}'_i$ to $\mathbf{F}\tilde{\mathbf{m}}_i$, its epipolar line. Likewise, $d(\tilde{\mathbf{m}}_i, \mathbf{F}^\top \tilde{\mathbf{m}}'_i)$ is the Euclidean distance of point $\tilde{\mathbf{m}}_i$ to $\mathbf{F}^\top \tilde{\mathbf{m}}'_i$. Therefore,

$$d(\tilde{\mathbf{m}}'_i, \mathbf{F}\tilde{\mathbf{m}}_i) = \frac{|\tilde{\mathbf{m}}'^{\top}_i \mathbf{F} \tilde{\mathbf{m}}_i|}{\|\mathbf{F}\tilde{\mathbf{m}}_i\|_2} \quad \text{and} \quad d(\tilde{\mathbf{m}}_i, \mathbf{F}^\top \tilde{\mathbf{m}}'_i) = \frac{|\tilde{\mathbf{m}}^\top \mathbf{F}^\top \tilde{\mathbf{m}}'_i|}{\|\mathbf{F}^\top \tilde{\mathbf{m}}'_i\|_2} \quad (3.7)$$

Note that the method of fundamental matrix estimation described here is not the only one. Several methods and parameterization of the fundamental matrix exist: iterative or non-iterative method, from 7 or more point correspondences, imposing or ignoring the rank-2 constraint of the fundamental matrix, normalizing or not the data, *etc.* Recently, Armangue proposed a comparison of all the different approaches [AS03].

3.1.3 Motion Recovery

When the fundamental matrix \mathbf{F} is estimated and the intrinsic parameters are known (after calibration), the essential matrix \mathbf{E} can be computed (see Equ. (3.3)). Then, the original motion (\mathbf{R}, \mathbf{t}) can be extracted from \mathbf{E} . The classical resolution of this problem is to decompose \mathbf{E} as described in [Zha96a]:

Starting from Equ. (3.4):

$$\begin{aligned}
 \text{Equ. (3.4)} &\Rightarrow \mathbf{E}^\top = \mathbf{R}^\top [\mathbf{t}]_\times^\top \\
 &\Leftrightarrow \mathbf{E}^\top \mathbf{t} = \mathbf{R}^\top \underbrace{[\mathbf{t}]_\times^\top \mathbf{t}}_0 \\
 &\Leftrightarrow \mathbf{E}^\top \mathbf{t} = \mathbf{0}
 \end{aligned} \tag{3.8}$$

Then, to recover the translation \mathbf{t} , we have to solve Equ. (3.8). This kind of problem is common in linear algebra and it is usually solved by minimizing $\|\mathbf{E}^\top \mathbf{t}\|_2^2$, subject to $\|\mathbf{t}\|_2 = 1$ (thus, the translation is found up to a scale factor). The solution is the unit eigenvector of $\mathbf{E}\mathbf{E}^\top$ corresponding to the smallest eigenvalue. This can be easily performed using Jacobi's method or a Singular Value Decomposition for example. However, because of the ambiguity of sign of the essential matrix, the sign of \mathbf{t} cannot be determined. This leads to four possible choices: $\{\mathbf{E}, \mathbf{t}\}$, $\{\mathbf{E}, -\mathbf{t}\}$, $\{-\mathbf{E}, \mathbf{t}\}$ and $\{-\mathbf{E}, -\mathbf{t}\}$. The ambiguity is removed during the 3D reconstruction process by choosing the only couple of solutions leading to points reconstructed in front of the imaging sensors.

For a given \mathbf{E} and \mathbf{t} , the rotation matrix \mathbf{R} can be found:

$$\text{Equ. (3.4)} \Rightarrow \mathbf{E} - [\mathbf{t}]_\times \mathbf{R} = \mathbf{0} \tag{3.9}$$

Like the previous problem, Equ. (3.9) is solved by minimizing $\|\mathbf{E} - [\mathbf{t}]_\times \mathbf{R}\|^2$, subject to $\mathbf{R}^\top \mathbf{R} = \mathbf{I}$ and $|\mathbf{R}| = 1$.

Moreover:

$$\begin{aligned}
 \mathbf{E} - [\mathbf{t}]_\times \mathbf{R} &= (\mathbf{E}\mathbf{R}^\top - [\mathbf{t}]_\times) \mathbf{R} \\
 \Rightarrow \|\mathbf{E} - [\mathbf{t}]_\times \mathbf{R}\|^2 &= \|(\mathbf{E}\mathbf{R}^\top - [\mathbf{t}]_\times)\|^2
 \end{aligned}$$

and the minimization becomes:

$$\min_{\mathbf{R}} \mathbf{C} \quad \text{with} \quad \mathbf{C} = \sum_{i=1}^3 \|\mathbf{R}\varepsilon_i - \tau_i\|_2^2$$

where ε_i et τ_i are respectively the i^{th} row vectors of \mathbf{E} and $[\mathbf{t}]_\times$.

Once again, the solution is the unit eigenvector of $\mathbf{C}^\top \mathbf{C}$ corresponding to the smallest eigenvalue. To solve it more easily, Zhang [Zha96b] proposed to use the quaternion representation of a rotation and to convert it to the rotation matrix \mathbf{R} (see Appendix C for details).

3.1.4 Specific Case of Parallel Projection

In the case of parallel projection, epipolar geometry is specifically called affine epipolar geometry in order to distinguish it from the general case of perspective projection. Contrary to the general epipolar geometry, epipolar lines in affine epipolar geometry are parallel because the center of projection is at infinity and the projections rays are parallel. However, every equation defined in the three previous sections 3.1.1, 3.1.2 and 3.1.3 are valid as well for parallel projection, the only difference is that most of them can be greatly simplified.

The fundamental matrix in the specific case of parallel projection is called affine fundamental matrix and defined as follows:

$$\mathbf{F}_{\parallel} = \begin{pmatrix} 0 & 0 & a \\ 0 & 0 & b \\ c & d & e \end{pmatrix}$$

where a, b, c, d and e are non-null. It is still of rank 2 but has only 4 degrees of freedom (it has 5 parameters but it is defined up to a scale factor). This simplifies the expression of the epipolar lines \mathbf{l}'_m and $\mathbf{l}_{m'}$ respectively associated to the points $\tilde{\mathbf{m}} = \begin{pmatrix} x & y & 1 \end{pmatrix}$ and $\tilde{\mathbf{m}}' = \begin{pmatrix} x' & y' & 1 \end{pmatrix}$:

$$\begin{aligned} \mathbf{l}'_m &= \mathbf{F}_{\parallel} \tilde{\mathbf{m}} = \begin{pmatrix} a & b & cx + dy + e \end{pmatrix}^{\top} \\ \text{and } \mathbf{l}_{m'} &= \mathbf{F}_{\parallel}^{\top} \tilde{\mathbf{m}}' = \begin{pmatrix} c & d & ax' + by' + e \end{pmatrix}^{\top} \end{aligned}$$

Estimation of the affine fundamental matrix is obviously simplified as well with respect to perspective projection. Using the same approach as in Section 3.1.2 and from n point matches, this leads to the same linear system $\mathbf{U}\mathbf{f}^{\top} = 0$ where $\mathbf{U} = \begin{pmatrix} \mathbf{u}_1 & \dots & \mathbf{u}_n \end{pmatrix}^{\top}$ but where \mathbf{u}_i and \mathbf{f} are defined more simply:

$$\begin{aligned} \text{Equ. (3.1)} &\Rightarrow ax'_i + by'_i + cx_i + dy_i + e = 0 \\ &\Leftrightarrow \mathbf{u}_i \mathbf{f}^{\top} = 0 \end{aligned}$$

where:

$$\begin{aligned} \mathbf{u}_i &= \begin{pmatrix} x'_i & y'_i & x_i & y_i & 1 \end{pmatrix} \\ \text{and } \mathbf{f} &= \begin{pmatrix} a & b & c & d & e \end{pmatrix} \end{aligned}$$

Here, the resolution of the linear system requires only 4 correspondences. Even if the *8-point algorithm* can be used again here, [HZ00] recommends to use instead the Gold

Standard algorithm which leads to greater performance with equal computational ease of implementation in the case of affine epipolar geometry.

The affine essential matrix is computed as the essential matrix:

$$\begin{aligned}
 \mathbf{E}_{//} &= \mathbf{A}_{//}^{\top} \mathbf{F}_{//} \mathbf{A}_{//} \\
 &= \begin{pmatrix} \alpha_x & 0 & 0 \\ s & \alpha_y & 0 \\ 0 & 0 & 1 \end{pmatrix} \mathbf{F}_{//} \begin{pmatrix} \alpha_x & s & 0 \\ 0 & \alpha_y & 0 \\ 0 & 0 & 1 \end{pmatrix} \\
 &= \begin{pmatrix} 0 & 0 & a\alpha_x \\ 0 & 0 & as + b\alpha_y \\ c\alpha_x & cs + d\alpha_y & e \end{pmatrix}
 \end{aligned}$$

Motion recovery from the affine essential matrix is not detailed here but the reader can refer to [HZ00] for further information.

3.2 Multiple View Geometry

Even if two images of an object are sufficient to be able to measure its 3D shape, it is common to acquire more images for more accuracy in the reconstruction. Moreover, it allows to have a complete view by taking images all around the object. Multiple view geometry [HZ00, FLP01] can be considered as a series of two-view geometry groups but it does not take into account specific cases and additional constraints appearing with three or more views. Trifocal and quadrifocal tensors are then introduced and can be considered as the extension of the fundamental matrix to three and four viewpoints. The previous section 3.1 about two-view geometry gives the necessary background for studying the multiple view geometry.

3.2.1 3-View Geometry

A geometry of three viewpoints can always be considered as the combination of two groups of two-view geometry. Then, using epipolar geometry properties and given correspondences in two images, it is possible to determine their correspondence in the third image as the intersection of the two epipolar lines (this method is known as epipolar transfer, see Fig. 3.4). However, this is not always possible and when the three centers of projection are nearly collinear or when the 3D point is close to the plane defined by the three centers (called trifocal plane), the epipolar transfer fails and the third correspondence must be determined by another way.

Fortunately, similarly to two-view geometry, there exist geometric relationships in three-view geometry which constrain points and lines correspondences over three images. This is represented by a $3 \times 3 \times 3$ homogeneous tensor described thereafter: the trifocal tensor. This third-order tensor has 27 parameters but only 18 are independent. Because we chose not to use the trifocal tensor for refining correspondences or for our 3D reconstruction process, the affine trifocal tensor (in parallel projection case) will not be discussed at all and the trifocal tensor will be presented only briefly. The literature is abundant on this topic and the reader can refer to [SA91, VL93, Sas94] for further information.

Trifocal Tensor

Similarly to two-view geometry, corresponding points and/or lines in three images are subject to different constraints. These are trilinear relationships (also called trilinearities), organized in the trifocal tensor τ . With two entities (point and line), five types of trilinearities exist: the point-point-point, the point-point-line, the point-line-point, the point-line-line and the line-line-line correspondence relations. The most useful are a) the point-point-point correspondence allowing to determine the location of a third corresponding point in the last view given two corresponding points in two views (point transfer)

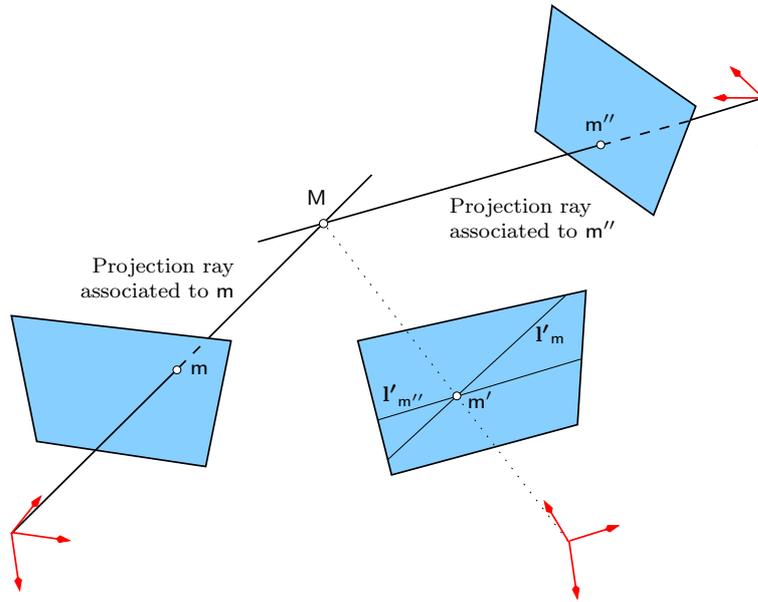


Fig. 3.4 — *Epipolar transfer: the correspondence m' in the second view is determined as the intersection of the epipolar line l'_m (image of the projection ray associated to m) and the epipolar line $l'_{m''}$ (image of the projection ray associated to m'').*

and b) the line-line-line correspondence allowing to determine the third corresponding line given two corresponding lines (line transfer):

- a) **Point transfer:** Given two correspondences m and m' , the third correspondence m'' can be determined using the trifocal tensor.

$$\forall i, j = 1, 2, 3, m''_i = m_i \sum_{k=1}^3 m_k \tau_{kjl} - m'_j \sum_{k=1}^3 m_k \tau_{kil} \quad [\text{TZ97}]$$

The nine different choices for the couple of values $\{i, j\}$ lead to nine expressions of point-point-point correspondence but only four are linearly independent.

- b) **Line transfer:** similarly, the third line correspondence l can be predicted from the two corresponding lines l' and l'' in the two other images:

$$\forall i = 1, 2, 3, l_i = \sum_{j=1}^3 \sum_{k=1}^3 l'_j l''_k \tau_{ijk} \quad [\text{TZ97}]$$

Three expressions of line-line-line correspondence exist but only two are linearly independent.

Trifocal Tensor Estimation

Analogous to the fundamental matrix in two-view geometry, the trifocal tensor is dependent of the intrinsic parameters and the motions between the three concerned viewpoints. Similarly to fundamental matrix estimation (see Section 3.1.2), it can be estimated directly from point correspondences and line correspondences as well and methods are close. The only difference and the main difficulty in the trifocal tensor estimation is the greater number of necessary constraints: while the fundamental matrix requires only one constraint (9 elements minus 1 scaling factor minus 7 degrees of freedom), the trifocal tensor estimation requires 8 geometric constraints (27 elements minus 1 scaling factor minus 18 independent parameters). These constraints are generally implicitly enforced by the parameterization of the trifocal tensor, the minimal parameterization having 18 parameters [FM95, Lav96, TZ97].

Several different algorithms exist to estimate the trifocal tensor [HZ00]. Linear methods require at least 7 points or 13 lines or a proper combination in three images whereas robust estimation based on RANSAC and some iterative methods minimizing an objective function can use only 6 point correspondences. Generally, more correspondences are available and the system is over-determined, leading to an optimal solution in the least-squares sense.

3.2.2 N-View Geometry

Recently, the quadrifocal tensor has been introduced and Hartley has proposed an algorithm for its computation [Har98]. This brings new constraints and should lead to better accuracy in 3D reconstruction than two- and three-view techniques. However, it is still early to fully understand and exploit its possibilities and its interest is currently more theoretical than practical.

No tensor exists for more than four views and whatever the number of viewpoints, the N-view geometry does not have multi-linear constraints for 5 views or more. It is expressed as a combination of two-, three- and four-view geometry using respectively the fundamental matrix, the trifocal tensor and the quadrifocal tensor. Actually, the quadrifocal tensor is rarely used practically and only the fundamental matrix and the trifocal tensor are used to find two-view and three-view correspondences to compute linear estimation of projection matrices for the 3D shape measurement process. Because image measurements are noisy, the estimation of the projection matrices is biased and it is always preferable to compute the 3D shape with a global optimization process such as the bundle adjustment, reestimating simultaneously the 3D points and the projection matrices.

3.3 Interest Point Detection and Matching

Fundamental matrix estimation requires a set of point correspondences (see Section 3.1.2) in each pair of images. Instead of selecting manually correspondences scattered in the images and in concern for the automation of the 3D reconstruction process, the correspondences will be found automatically. The idea is to use an interest point detector to extract remarkable points in each image in order to match them easily throughout the image sequence. The correspondences can then be gathered for each pair of images and the corresponding fundamental matrix can be estimated. A very good survey of different strategies from the point extraction to the epipolar geometry estimation is proposed by Vincent [VL01]. Note that the same technique would be used for the trifocal tensor estimation (the correspondences would be gathered by triplet of images) but we chose to not use it in the reconstruction process.

3.3.1 Interest Point Detection

Numerous interest point detectors exist in the literature: not only the well-known Harris [HS88], SUSAN [SB97] and Canny [Can86] edge detectors, but also Horaud *et al.* [HSV90], Heitger *et al.* [HRVDH⁺92], Cottier [Cot94] and Forstner [For94]. Because results depend widely on the detector and the algorithm used, a comparison is necessary to find the most suitable for our specific needs. Detectors are most often compared with respect to a localization accuracy criterion but this is not significant for us. Indeed, the aim of the interest point detector for us is not to locate precisely a feature point in the image (like a corner) but to find a point very characteristic which will be repeated in the most possible images in the sequence. The motion between different viewpoints is seldom only a translation and the interest detector must extract points independently of the rotation and the scale of the images. Moreover, brightness and contrast in SEM images are far from being constant in an image sequence and therefore, interest points must be invariant to these parameters. At last, interest points also need to be regularly scattered in the whole image because we notice that in practice, the epipolar geometry estimated is very badly estimated if the point correspondences are all in a same area of the image.

Schmid *et al.* [SMB00] compare several different detectors and use different criteria to evaluate them. One of them is very interesting for us: the repeatability rate. Indeed, this criterion compares the stability of the detector with respect to different images of a same scene under varying viewing conditions. According to the authors, the best detector evaluated with the repeatability criterion is an improved version of Harris detector.

Like several interest point detectors, Harris detector uses an intensity based method (opposed to contour based and parametric model based methods), *i.e.* it uses the grey levels of the image to measure the *cornerness* (also called “corner response”) of each point.

The cornerness is a value which permits to detect interest points by quantifying their “interest”. Its computation is based on the auto-correlation matrix Ω , approximation of the Hessian matrix of the auto-correlation function. This function characterizes how the intensity values change in an area defined around the point (x, y) . It is computed by convolving the intensity image derivatives with a Gaussian filter:

$$\Omega(x, y) = \frac{\exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)}{2\pi\sigma^2} \otimes \begin{pmatrix} \left(\frac{\partial I}{\partial x}\right)^2 & \frac{\partial I}{\partial x} \frac{\partial I}{\partial y} \\ \frac{\partial I}{\partial x} \frac{\partial I}{\partial y} & \left(\frac{\partial I}{\partial y}\right)^2 \end{pmatrix}$$

where $I(x, y)$ is the grey level intensity and \otimes the convolution product.

If both eigenvalues of the matrix Ω are high, then a small motion of the local patch in any direction will cause a significant change in grey level and this indicates that the point is a feature one. That is why the cornerness depends on the determinant of Ω , which is proportional of the product of the two eigenvalues:

$$c(x, y) = |\Omega(x, y)| - 0.04 \times \text{trace}(\Omega(x, y))^2$$

Cornerness is then computed for each image point and local maxima of the function are selected as interest points. Sub-pixel accuracy can be obtained using a quadratic fitting of the neighborhood of the local maxima. A threshold is eventually used to keep only the best of the selected points.

However, this detector is not directly usable for us because of the additional constraint of the distribution of the points in the image. The classical implementation of the Harris detector leads to points often concentrated in the same area of the image and therefore, this implementation has been modified to obtain points distributed in the whole image: instead of selecting an unknown number of points in the image with respect to a threshold, the image is divided in small areas (bucketing technique) in which we select the best points. Obviously, a threshold is used as well to prevent points that should not be repeatable.

3.3.2 Interest Point Matching

At this point, we have now one set of feature points per image and in order to estimate the epipolar geometry between each pair of images (see Section 3.1.2), the interest points of the different sets have to be matched between them. Here, a zero mean normalized cross correlation (ZNCC) function will be used but for an obvious matter of speed, a point of a given set cannot be compared with each point of the other set. To optimize this process, a point is only compared with points satisfying the two following criteria:

- small distance to the current point (actually, the correspondence point coordinates have to be in a rectangle area defined around the current point),

- good cornerness similarity [JL01] with the current point.

In practice, we usually fix the similarity at 50% because a greater value eliminates good matching and the size of the area depends on the motion between the viewpoints but 25% of the image size is often a good compromise.

Eventually, in order to prevent false matching, a robust filtering process (Least Median of Squares technique [RL87]) detects and removes the outliers. The Least Median of Squares (LMS) estimator is a statistical approach which, as the name indicates, minimizes the median value of the squared residuals. The complete algorithm of outliers filtering is composed as follows:

1. For $j = 1 \dots r$, repeat:
 - a) Eight point correspondences are randomly chosen in the entire set of available n correspondences¹.
 - b) A fundamental matrix \mathbf{F}_j is estimated using only these 8 point correspondences and the *8-point algorithm*.
 - c) For each pair of correspondences $\{\tilde{\mathbf{m}}_i, \tilde{\mathbf{m}}'_i\}, i = 1 \dots n$ in the set of data, the squared residual $\epsilon_{i,j}^2$ is computed: $\epsilon_{i,j}^2 = d^2(\tilde{\mathbf{m}}'_i, \mathbf{F}_j \tilde{\mathbf{m}}_i) + d^2(\tilde{\mathbf{m}}_i, \mathbf{F}_j^\top \tilde{\mathbf{m}}'_i)$ where d represents the Euclidean distance function.
 - d) The median value m_j of all n squared residuals $\epsilon_{i,j}$ is computed.
2. The fundamental matrix $\mathbf{F}_{\hat{j}}$ is selected where \hat{j} is the value of $j = 1 \dots r$ for which m_j is the minimal median value.
3. Point correspondences are considered as outliers if their associated squared residual $\epsilon_{i,j}$ is greater than the threshold $(2.5 \hat{\sigma}^2)$ where $\hat{\sigma} = \frac{1.4826(1+5/(n-8))}{\sqrt{m_j}}$ is the robust standard deviation (see [RL87, Zha96b] for details).

The number r of repetitions of steps a) to d) is given in [Zha96b]: assuming a fraction of outliers f in the set of correspondences, the probability that at least one of the r subsamples of 8 points is good is $p = 1 - (1 - (1 - f)^8)^r$. For a fraction of outliers $f = 20\%$ and with an objective of probability $p = 99\%$, the number of necessary subsamples is $r = 26$.

¹Actually, if the 8 point correspondences were really randomly chosen, there would be a chance that the correspondences be close in the image and the fundamental matrix would be badly estimated. For this reason, one of the images is divided into buckets (typically 8×8) and only one point per bucket is allowed in this image.

3.4 Triangulation

When the imaging sensor is calibrated and the relative position and orientation are recovered, the last step to obtain 3D shape measurement is the triangulation. Without the determination of the intrinsic parameters, projection matrices cannot be computed but a 3D projective reconstruction is still possible (up to a projective transformation). It can be sufficient for some applications but not for ours and intrinsic parameters are therefore necessary to upgrade the reconstruction to a metric one (up to a scale factor). The metric reconstruction is adequate for strain measurements (Chapter 4) but for most of 3D shape measurement applications, the metric reconstruction needs to be upgraded to Euclidean reconstruction by finally determining the missing scale factor (see Section 2.2.1 and [Fau95] for the stratification of geometry). This is generally achieved using the knowledge of a distance in the scene or a known accurate displacement between two viewpoints.

The first section 3.4.1 presents the equations to solve to recover the 3D information from the projections matrices and measure points. Note that we chose to not compute trifocal tensors and only fundamental matrices are used to estimate the projection matrices, even with 3 views and more. However, once the two first projection matrices are estimated, the third and following projection matrices cannot be estimated directly because they depend on the frame defined by the first two. A specific method of computation is therefore necessary for more than 2 views and details are given in Section 3.4.1. Section 3.4.2 presents different techniques to solve the linear equation system coming from the triangulation. The last section (Section 3.4.3) treats the specific case of parallel projection.

3.4.1 3D Metric Reconstruction from Multi-View Images

As illustrated in Fig. 3.5, for each pair of viewpoints (i, j) , the rigid-body transformation between \mathcal{R}_i^ξ and \mathcal{R}_j^ξ is denoted:

$$\mathbf{T}_{i \rightarrow j} = \begin{pmatrix} \mathbf{R}_{i,j} & \mathbf{t}_{i,j} \\ \mathbf{0}_3^\top & 1 \end{pmatrix} \quad (3.10)$$

After calibration and epipolar geometry estimation, the intrinsic matrix \mathbf{K} and motions $\mathbf{T}_{i \rightarrow j}$ between each couple of viewpoints are estimated. Actually, we saw in Section 3.1.3 that rotation matrices $\mathbf{R}_{i,j}$ are completely recovered but each translation vector $\mathbf{t}_{i,j}$ is only recovered up to a scale factor $\lambda_{i,j}$. If we choose the i^{th} view as the reference one (*i.e.* the i^{th} imaging sensor coordinate system is set as world coordinate system), the projection matrix corresponding to the view j is:

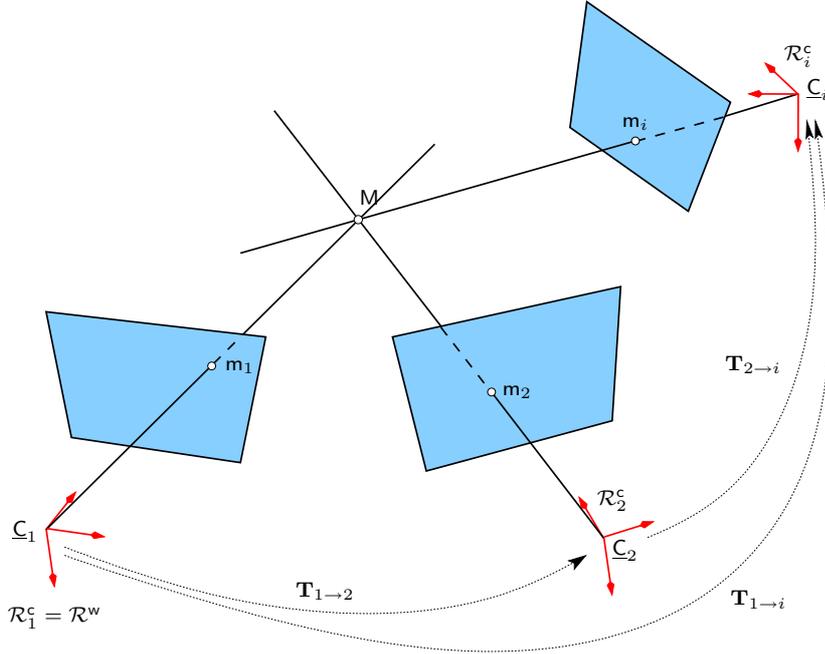


Fig. 3.5 — 3D reconstruction from multiple views: all the projection rays are intersecting in space at the point M . Here, the first view is chosen as the reference one and its imaging sensor coordinate system is set as world coordinate system.

$$\begin{aligned}
 \mathbf{H}^j &= \mathbf{K} \begin{pmatrix} \mathbf{R}_{i,j} & (\lambda_{i,j} \mathbf{t}_{i,j}) \\ \mathbf{0}_3^\top & 1 \end{pmatrix} \\
 &= \mathbf{A} [\mathbf{R}_{i,j} \ (\lambda_{i,j} \mathbf{t}_{i,j})] \\
 &= \begin{pmatrix} h_{11}^j & h_{12}^j & h_{13}^j & h_{14}^j \\ h_{21}^j & h_{22}^j & h_{23}^j & h_{24}^j \\ h_{31}^j & h_{32}^j & h_{33}^j & h_{34}^j \end{pmatrix}
 \end{aligned} \tag{3.11}$$

To highlight the influence of $\lambda_{i,j}$ on the last column of \mathbf{H}^j , the elements of this column will be denoted $h_{\{1,2,3\}4}^j = \lambda_{i,j} \mathring{h}_{\{1,2,3\}4}^j$ until the end of this section.

Note that the projection matrix associated to a view j can be determined even if this view does not have any point correspondences with the reference view. The view j just requires to have point correspondences² with one or more intermediate views, having themselves point correspondences with the reference view (for instance, the projection matrix in view 4 can be computed if the view 4 has point correspondences with the view

²Actually, it requires to have at least 7 point correspondences so that epipolar geometry can be estimated.

3, the view 3 having point correspondences with the view 2 and this latter having point correspondences with the reference view 1, see Fig. 3.6).

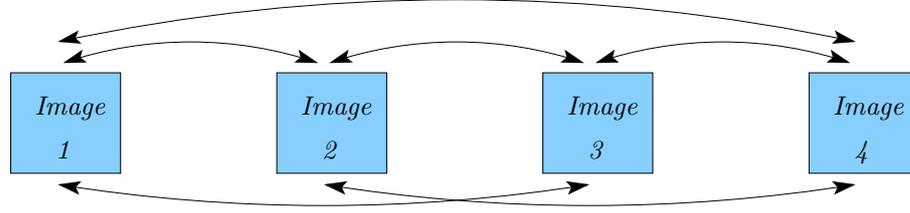


Fig. 3.6 — Possibilities of organization of 4 views: even if there is not enough point correspondences between the view 1 and 4, projection matrix of view 4 may be computed using its relationships with the view 2 or 3.

This is the big advantage of the multiple view reconstruction using a single sensor and the videogrammetry technique: any point of the specimen can be reconstructed on condition that it is visible in at least two views. Therefore, a complete 3D reconstruction of the specimen can be obtained by acquiring images all around the object (see Fig. 3.7) and so that the views are overlapping in order to have points visible in more than one view.

We remind that if a perspective projection model is assumed for the imaging sensor, the following relationship exists:

$$\forall i = 1 \dots n \text{ (number of points), } j = 1 \dots m \text{ (number of views),}$$

$$\tilde{\mathbf{m}}_i^j \cong \mathbf{H}_\perp^j \tilde{\mathbf{M}}_i \quad (3.12)$$

where $\tilde{\mathbf{m}}_i^j = \begin{pmatrix} x_i^j & y_i^j & 1 \end{pmatrix}^\top$ is the projected point of $\tilde{\mathbf{M}}_i = \begin{pmatrix} X_i & Y_i & Z_i & 1 \end{pmatrix}^\top$ in the j^{th} sensor plane (see Fig. 3.5). For convenience, \mathbf{H}_\perp^j is simply denoted \mathbf{H}^j in the following sections.

3D reconstruction from two views

Without loss of generality, let's assume that the first viewpoint is chosen as reference view. The projection matrices are then expressed as follows (see Equ. (3.11)):

$$\mathbf{H}^1 = \mathbf{A}_1 [\mathbf{I}_3 \ \mathbf{0}_3]$$

$$\mathbf{H}^2 = \mathbf{A}_2 [\mathbf{R}_{1,2} \ (\lambda_{1,2} \mathbf{t}_{1,2})]$$

$$= \begin{pmatrix} h_{11}^1 & h_{12}^1 & h_{13}^1 & 0 \\ h_{21}^1 & h_{22}^1 & h_{23}^1 & 0 \\ h_{31}^1 & h_{32}^1 & h_{33}^1 & 0 \end{pmatrix} \quad = \begin{pmatrix} h_{11}^2 & h_{12}^2 & h_{13}^2 & \lambda_{1,2} \mathring{h}_{14}^2 \\ h_{21}^2 & h_{22}^2 & h_{23}^2 & \lambda_{1,2} \mathring{h}_{24}^2 \\ h_{31}^2 & h_{32}^2 & h_{33}^2 & \lambda_{1,2} \mathring{h}_{34}^2 \end{pmatrix}$$

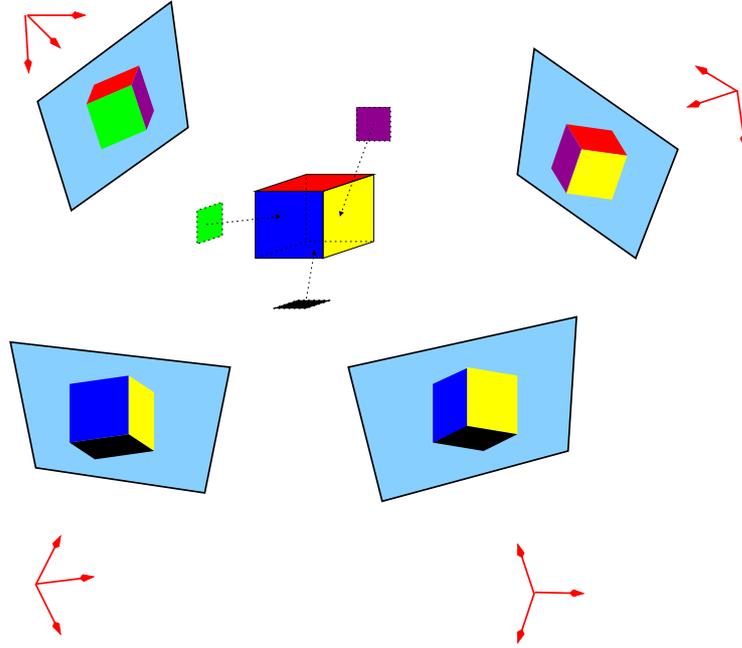


Fig. 3.7 — *Example of 4 views all around an object: in order to reconstruct the complete shape of the specimen, multiple images are acquired from different viewpoints. However, the points meant to be reconstructed need also to be visible in at least two views. Then, in this example, the green face of the cube cannot be reconstructed.*

From Equ. (3.12), we can write $\tilde{\mathbf{m}}^1 \cong \mathbf{H}^1 \tilde{\mathbf{M}}$ and its correspondence in the second view $\tilde{\mathbf{m}}^2 \cong \mathbf{H}^2 \tilde{\mathbf{M}}$ and deduce:

$$\begin{aligned} x^1 &= \frac{h_{11}^1 X + h_{12}^1 Y + h_{13}^1 Z}{h_{31}^1 X + h_{32}^1 Y + h_{33}^1 Z} & y^1 &= \frac{h_{21}^1 X + h_{22}^1 Y + h_{23}^1 Z}{h_{31}^2 X + h_{32}^2 Y + h_{33}^2 Z} \\ x^2 &= \frac{h_{11}^2 X + h_{12}^2 Y + h_{13}^2 Z + \lambda_{1,2} \overset{\circ}{h}_{14}^2}{h_{31}^2 X + h_{32}^2 Y + h_{33}^2 Z + \lambda_{1,2} \overset{\circ}{h}_{34}^2} & y^2 &= \frac{h_{21}^2 X + h_{22}^2 Y + h_{23}^2 Z + \lambda_{1,2} \overset{\circ}{h}_{24}^2}{h_{31}^2 X + h_{32}^2 Y + h_{33}^2 Z + \lambda_{1,2} \overset{\circ}{h}_{34}^2} \end{aligned}$$

This can be expressed as an over-determined linear system (4 equations for 3 unknowns):

$$\begin{cases} (x^1 h_{31}^1 - h_{11}^1) X + (x^1 h_{32}^1 - h_{12}^1) Y + (x^1 h_{33}^1 - h_{13}^1) Z = 0 \\ (y^1 h_{31}^1 - h_{21}^1) X + (y^1 h_{32}^1 - h_{22}^1) Y + (y^1 h_{33}^1 - h_{23}^1) Z = 0 \\ (x^2 h_{31}^2 - h_{11}^2) X + (x^2 h_{32}^2 - h_{12}^2) Y + (x^2 h_{33}^2 - h_{13}^2) Z = \lambda_{1,2} (\overset{\circ}{h}_{14}^2 - x^2 \overset{\circ}{h}_{34}^2) \\ (y^2 h_{31}^2 - h_{21}^2) X + (y^2 h_{32}^2 - h_{22}^2) Y + (y^2 h_{33}^2 - h_{23}^2) Z = \lambda_{1,2} (\overset{\circ}{h}_{24}^2 - y^2 \overset{\circ}{h}_{34}^2) \end{cases} \quad (3.13)$$

Here, we clearly notice that this system will be solved up to the scale factor $\lambda_{1,2}$ and the

reconstructed 3D point $\begin{pmatrix} X & Y & Z \end{pmatrix}^\top$ will be found up to this same factor. The solving of this equation system will be discussed in Section 3.4.2.

3D reconstruction from more than 2 views

With 3 or more images and because we do not use the trifocal tensor, the projections matrices cannot be computed directly from the estimated translations. Indeed, if the same approach is kept and if we re-wrote the linear system of equations (3.13) with m views, $m-1$ different scale factors will be obtained due to each scale factor in estimated translations. Actually, we know that geometric constraints exist and all scale factors can be formulated in function of a global scale.

Without loss of generality, the first view is set as reference view and all scale factors are expressed with respect to $\lambda_{1,2}$. Then, for any view j :

$$\begin{aligned} \mathbf{T}_{1 \rightarrow j} &= \mathbf{T}_{2 \rightarrow j} \cdot \mathbf{T}_{1 \rightarrow 2} \\ \begin{pmatrix} \mathbf{R}_{1,j} & \mathbf{t}_{1,j} \\ \mathbf{0}_3^\top & 1 \end{pmatrix} &= \begin{pmatrix} \mathbf{R}_{2,j} & \mathbf{t}_{2,j} \\ \mathbf{0}_3^\top & 1 \end{pmatrix} \cdot \begin{pmatrix} \mathbf{R}_{1,2} & \mathbf{t}_{1,2} \\ \mathbf{0}_3^\top & 1 \end{pmatrix} \\ \implies \mathbf{t}_{1,j} &= \mathbf{R}_{2,j} \mathbf{t}_{1,2} + \mathbf{t}_{2,j} \end{aligned} \quad (3.14)$$

Actually, we do not know $\mathbf{t}_{1,j}$, $\mathbf{t}_{1,2}$ and $\mathbf{t}_{2,j}$ but only $\tilde{\mathbf{t}}_{1,j} = \lambda_{1,j} \mathbf{t}_{1,j}$, $\tilde{\mathbf{t}}_{1,2} = \lambda_{1,2} \mathbf{t}_{1,2}$ and $\tilde{\mathbf{t}}_{2,j} = \lambda_{2,j} \mathbf{t}_{2,j}$, then:

$$\begin{aligned} (3.14) \implies \mathbf{t}_{2,j} \times \mathbf{t}_{1,j} &= \mathbf{t}_{2,j} \times \mathbf{R}_{2,j} \mathbf{t}_{1,2} + \underbrace{\mathbf{t}_{2,j} \times \mathbf{t}_{2,j}}_{=\mathbf{0}_3} \\ \implies \frac{\tilde{\mathbf{t}}_{2,j}}{\lambda_{2,j}} \times \frac{\tilde{\mathbf{t}}_{1,j}}{\lambda_{1,j}} &= \frac{\tilde{\mathbf{t}}_{2,j}}{\lambda_{2,j}} \times \mathbf{R}_{2,j} \frac{\tilde{\mathbf{t}}_{1,2}}{\lambda_{1,2}} \\ \implies \lambda_{1,2} (\tilde{\mathbf{t}}_{2,j} \times \tilde{\mathbf{t}}_{1,j}) &= \lambda_{1,j} (\tilde{\mathbf{t}}_{2,j} \times \mathbf{R}_{2,j} \tilde{\mathbf{t}}_{1,2}) \\ \implies \lambda_{1,2} (\tilde{\mathbf{t}}_{2,j} \times \tilde{\mathbf{t}}_{1,j})^\top (\tilde{\mathbf{t}}_{2,j} \times \tilde{\mathbf{t}}_{1,j}) &= \lambda_{1,j} (\tilde{\mathbf{t}}_{2,j} \times \tilde{\mathbf{t}}_{1,j})^\top (\tilde{\mathbf{t}}_{2,j} \times \mathbf{R}_{2,j} \tilde{\mathbf{t}}_{1,2}) \\ \implies \lambda_{1,2} &= \lambda_{1,j} \frac{(\tilde{\mathbf{t}}_{2,j} \times \tilde{\mathbf{t}}_{1,j})^\top (\tilde{\mathbf{t}}_{2,j} \times \mathbf{R}_{2,j} \tilde{\mathbf{t}}_{1,2})}{\|\tilde{\mathbf{t}}_{2,j} \times \tilde{\mathbf{t}}_{1,j}\|_2^2} \end{aligned} \quad (3.15)$$

Therefore, each scale factor can be expressed as

$$\lambda_{1,j} = \frac{1}{\Lambda_j} \lambda_{1,2} \text{ where } \Lambda_j = \frac{(\tilde{\mathbf{t}}_{2,j} \times \tilde{\mathbf{t}}_{1,j})^\top (\tilde{\mathbf{t}}_{2,j} \times \mathbf{R}_{2,j} \tilde{\mathbf{t}}_{1,2})}{\|\tilde{\mathbf{t}}_{2,j} \times \tilde{\mathbf{t}}_{1,j}\|_2^2}$$

and before computing projection matrices, translation vectors have to be scaled in order to be expressed with the same global scale. By this way, $\lambda_{1,2}$ remains the only unknown scale factor in the equation system.

$$\mathbf{A} = \begin{pmatrix} x^1 h_{31}^1 - h_{11}^1 & x^1 h_{32}^1 - h_{12}^1 & x^1 h_{33}^1 - h_{13}^1 \\ y^1 h_{31}^1 - h_{21}^1 & y^1 h_{32}^1 - h_{22}^1 & y^1 h_{33}^1 - h_{23}^1 \\ \vdots & \vdots & \vdots \\ x^j h_{31}^j - h_{11}^j & x^j h_{32}^j - h_{12}^j & x^j h_{33}^j - h_{13}^j \\ \vdots & \vdots & \vdots \\ y^m h_{31}^m - h_{21}^m & y^m h_{32}^m - h_{22}^m & y^m h_{33}^m - h_{23}^m \end{pmatrix}, \quad \mathbf{M} = \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \quad \text{and} \quad \mathbf{b} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ h_{14}^j - x^j h_{34}^j \\ \vdots \\ h_{24}^m - y^m h_{34}^m \end{pmatrix}$$

Since \mathbf{A} is not square and then not invertible, the solution of this equation is: $\mathbf{M} = \mathbf{A}^\dagger \mathbf{b}$ where $\mathbf{A}^\dagger = (\mathbf{A}^\top \mathbf{A})^{-1} \mathbf{A}^\top$ is the Moore-Penrose inverse matrix [Pen55] (pseudo-inverse matrix) of \mathbf{A} . The solution found by this way is optimal in a least-squares sense.

Non-Iterative Method: Null-Space Method: Another matrix expression of the linear system (3.16) is $\mathbf{A}\tilde{\mathbf{M}} = \mathbf{0}$ where:

$$\mathbf{A} = \begin{pmatrix} x^1 h_{31}^1 - h_{11}^1 & x^1 h_{32}^1 - h_{12}^1 & x^1 h_{33}^1 - h_{13}^1 & 0 \\ y^1 h_{31}^1 - h_{21}^1 & y^1 h_{32}^1 - h_{22}^1 & y^1 h_{33}^1 - h_{23}^1 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ x^j h_{31}^j - h_{11}^j & x^j h_{32}^j - h_{12}^j & x^j h_{33}^j - h_{13}^j & x^j h_{34}^j - h_{14}^j \\ \vdots & \vdots & \vdots & \vdots \\ y^m h_{31}^m - h_{21}^m & y^m h_{32}^m - h_{22}^m & y^m h_{33}^m - h_{23}^m & y^m h_{34}^m - h_{24}^m \end{pmatrix} \quad \text{and} \quad \tilde{\mathbf{M}} = \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

$\tilde{\mathbf{M}}$ is here defined as the null-space of matrix \mathbf{A} and can be determined either by Singular Value Decomposition of \mathbf{A} or by Jacobi's method ($\tilde{\mathbf{M}}$ is the eigenvector associated to the smallest eigenvalue of $\mathbf{A}^\top \mathbf{A}$).

Iterative Method: The two previous methods lead to an algebraic solution which has no actual physical interpretation. This is why it is preferable to carry out an iterative method which minimizes a physically meaningful criterion. The quantity chosen to be minimized is the sum of the Euclidean distances between the 2D image points and the reprojections of the reconstructed 3D point (reprojection error):

$$\min_{\tilde{\mathbf{M}}} \sum_{j=1}^m \|\epsilon^j\|_2^2 \quad \text{where} \quad \epsilon^j = \tilde{\mathbf{m}}^j - \mathbf{H}^j \tilde{\mathbf{M}}$$

The minimization can be achieved with the Levenberg-Marquardt method where the Jacobian vector would be:

$$\frac{\partial \epsilon^j}{\partial \tilde{\mathbf{M}}} = \frac{1}{\left(\mathbf{h}_3^j \tilde{\mathbf{M}}\right)^2} \begin{pmatrix} h_{11}^j \cdot \mathbf{h}_3^j \tilde{\mathbf{M}} - h_{31}^j \cdot \mathbf{h}_1^j \tilde{\mathbf{M}} & h_{12}^j \cdot \mathbf{h}_3^j \tilde{\mathbf{M}} - h_{32}^j \cdot \mathbf{h}_1^j \tilde{\mathbf{M}} & h_{13}^j \cdot \mathbf{h}_3^j \tilde{\mathbf{M}} - h_{33}^j \cdot \mathbf{h}_1^j \tilde{\mathbf{M}} \\ h_{21}^j \cdot \mathbf{h}_3^j \tilde{\mathbf{M}} - h_{31}^j \cdot \mathbf{h}_2^j \tilde{\mathbf{M}} & h_{22}^j \cdot \mathbf{h}_3^j \tilde{\mathbf{M}} - h_{32}^j \cdot \mathbf{h}_2^j \tilde{\mathbf{M}} & h_{23}^j \cdot \mathbf{h}_3^j \tilde{\mathbf{M}} - h_{33}^j \cdot \mathbf{h}_2^j \tilde{\mathbf{M}} \end{pmatrix}$$

where $\mathbf{h}_k^j = (h_{k1}^j \ h_{k2}^j \ h_{k3}^j \ h_{k4}^j)$ is the k^{th} row of \mathbf{H}^j .

Extension to Bundle Adjustment

The three previous methods are generally used when only one 3D point has to be reconstructed or for determining initial guesses for the current described method. Indeed, in order to improve the quality of the 3D reconstruction, a global optimization is preferable and the previous method can be extended to a bundle adjustment where all the 3D points and all the projection matrices are reestimated simultaneously. In other words, the bundle adjustment considers all the views and all the 3D points in a global way and minimizes, as usual, the sum of magnitudes of reprojection errors. This is the final step of the 3D shape measurement.

Note that a 3D point is not always projected in all views but the more views it is visible in, the more accurate will be its estimation (we remind that at least two point correspondences, *i.e.* the 3D point projected in two views, are necessary to reconstruct the point in 3D). This is why the images are acquired not only all around the object but also so that the views are overlapping. By this way, any point of the object can be reconstructed (contrary to stereo-vision applications where only a part of the object is observed) and moreover, these points are accurately reconstructed because visible in many views.

Even if the projection matrices are entirely reestimated in some approaches, we prefer to fix the intrinsic parameters for the bundle adjustment (see Fig. 3.8) because they have already been accurately estimated during the calibration phase. Moreover, in our case the intrinsic parameters are known and constant through all the viewpoints (since we have a unique imaging sensor) and allowing the entire projection matrices (and then indirectly the intrinsic parameters) to be reestimated for each viewpoint may cause an error on the estimation of the extrinsic parameters. Therefore, it is better to parameterize differently the projection matrices and the intrinsic matrix is separated from the extrinsic matrix for each projection matrix: \mathbf{H}^j is expressed as a combination of a unique \mathbf{K} and one matrix \mathbf{T}^j . By this way, we can more easily ensure the constancy of the intrinsic matrix during the optimization process. The whole process and the number of constraints to enforce is already described in detail in Section 2.3 and in Appendix A.

Note that the bundle adjustment was already used in the calibration phase for the projection model parameters estimation (see Section 2.3.1, Step 3c. and particularly Fig. 2.21). Interestingly, even if the calibration and 3D shape measurement are different matters, an

analogy can be established between the use of the bundle adjustment in both cases by comparing Fig. 2.21 and Fig. 3.8:

- involved parameters are exactly the same in both cases: measure points from acquired images, 3D shape, projection model parameters and distortion parameters.
- the measure points used as input come from images of a calibration target for the calibration and they come from images of the object to reconstruct for the 3D shape measurement.
- the bundle adjustment outputs a 3D shape in both cases but it is only the interesting information for the 3D shape measurement.
- projection model parameters needs to be given as input in both cases but they are initial guesses to be refined for the calibration whereas they will remain fixed for the 3D shape measurement.

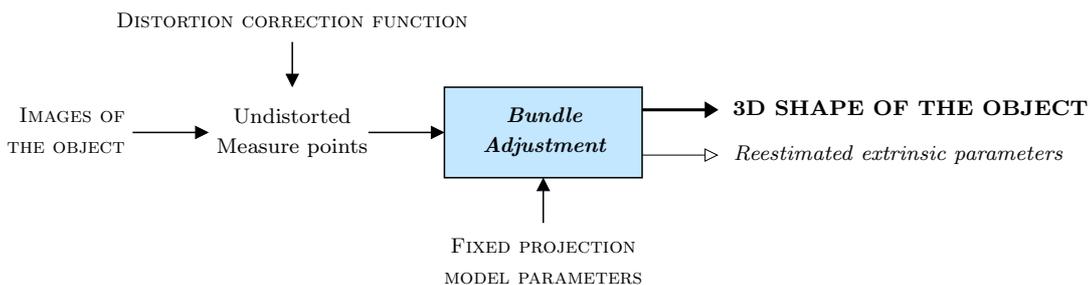


Fig. 3.8 — *The bundle adjustment is used for the calibration (see Fig. 2.21) as well as the 3D shape measurement. Involved parameters are similar in both cases and only the role of the parameters differs.*

3.4.3 3D Reconstruction using Parallel Projection Imaging Sensors

When the parallel projection is assumed and without intrinsic parameters, the 3D reconstruction can only be determined up to an affine transformation. It is better than the projective ambiguity obtained with perspective projection imaging sensors but it is still insufficient. If the 3 intrinsic parameters are known, the reconstruction is upgraded to a metric one, similarly to the perspective projection case.

The 3D reconstruction with parallel projection imaging sensors is achieved using the “factorization algorithm” introduced by Tomasi and Kanade [TK92] and described thereafter. Contrary to the bundle adjustment algorithm in the perspective projection case, this

algorithm requires that the n 3D points meant to be reconstructed has projected points in the all m views.

Equ. (2.4) of the general projective transformation can be applied to parallel projection:

$$\begin{aligned} \forall i = 1 \dots n, j = 1 \dots m, \\ \tilde{\mathbf{m}}_i^j \cong \mathbf{H}_{\parallel}^j \tilde{\mathbf{M}}_i \end{aligned} \quad (3.17)$$

where $\tilde{\mathbf{m}}_i^j = \begin{pmatrix} x_i^j & y_i^j & 1 \end{pmatrix}^\top$ is the i^{th} 2D projected image point of the 3D point $\mathbf{M}_i = \begin{pmatrix} X_i & Y_i & Z_i & 1 \end{pmatrix}^\top$ in the view j . For convenience, until the end of this section, the 2D and 3D points will be denoted in inhomogeneous coordinates.

Moreover, the translation vector is separated from the projection matrix \mathbf{H}_{\parallel}^j . Then, the equation of the parallel projection Equ. (3.17) becomes:

$$\mathbf{m}_i^j = \mathbf{N}^j \mathbf{M}_i + \mathbf{t}^j$$

where \mathbf{N}^j is a 2×3 matrix and \mathbf{t}^j a 2D vector both extracted from \mathbf{H}_{\parallel}^j . Obviously, the projection points \mathbf{m}_i are biased by measurement noise and are not perfectly equal to $\mathbf{N}^j \mathbf{M}_i + \mathbf{t}^j$. Similarly to the bundle adjustment method in the perspective projection case, the 3D points \mathbf{M}_i are then estimated by minimizing the sum of magnitudes of reprojection errors (distance between the measured image points \mathbf{m}_i^j and the estimated projected points equal to $\mathbf{N}^j \mathbf{M}_i + \mathbf{t}^j$):

$$\min_{\mathbf{N}^j, \mathbf{M}_i, \mathbf{t}^j} \sum_i \sum_j \|\epsilon_i^j\|_2^2 \quad \text{where } \epsilon_i^j = \mathbf{m}_i^j - \mathbf{N}^j \mathbf{M}_i - \mathbf{t}^j \quad (3.18)$$

The translation vector \mathbf{t}^j can be removed from the minimization by changing the coordinate system such that the centroid of the points is at the origin [MZ92]. This statement is explained by the fact that for a parallel projection imaging system, the centroid of a set of 3D points is projected to the centroid of the corresponding projected image points. Every measure point \mathbf{m}_i^j is then translated so that the centroid of the n points in each view j is at the origin (leading to $\mathbf{t}^j = 0$):

$$\check{\mathbf{m}}_i^j = \mathbf{m}_i^j - \frac{1}{n} \sum_{i=1}^n \mathbf{m}_i^j$$

This is why it is necessary that the n 3D points are projected in all views (n must be the same for the m images). The minimization (3.18) turns into:

$$\min_{\mathbf{N}^j, \mathbf{M}_i} \sum_i \sum_j \|\check{\mathbf{m}}_i^j - \mathbf{N}^j \mathbf{M}_i\|_2^2 \quad (3.19)$$

This can then be expressed in matrix form by introducing the $2m \times n$ measurement matrix \mathbf{W} [HZ00]:

$$\mathbf{W} = \begin{pmatrix} \tilde{x}_1^1 & \tilde{x}_2^1 & \dots & \tilde{x}_n^1 \\ \tilde{y}_1^1 & \tilde{y}_2^1 & \dots & \tilde{y}_n^1 \\ \tilde{x}_1^2 & \tilde{x}_2^2 & \dots & \tilde{x}_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{x}_1^m & \tilde{x}_2^m & \dots & \tilde{x}_n^m \\ \tilde{y}_1^m & \tilde{y}_2^m & \dots & \tilde{y}_n^m \end{pmatrix}$$

and writing:

$$(3.19) \Rightarrow \mathbf{W} = \begin{pmatrix} \mathbf{N}^1 \\ \mathbf{N}^2 \\ \vdots \\ \mathbf{N}^m \end{pmatrix} \begin{pmatrix} \mathbf{M}_1 & \mathbf{M}_2 & \dots & \mathbf{M}_n \end{pmatrix} \quad (3.20)$$

In the presence of noise, Equ. (3.20) may not be satisfied [HZ00] and the solution consists in determining the matrix $\hat{\mathbf{W}}$ which minimizes the Frobenius norm of $\|\mathbf{W} - \hat{\mathbf{W}}\|$. This is achieved easily using a Singular Value Decomposition of \mathbf{W} .

3.5 Applications

3D reconstruction using a single sensor and the videogrammetry technique is very interesting not only for SEM systems (where stereo-vision is not possible) but also for every imaging systems used in non-contacting surface measurements ([CLVWdV02] for example). It has become a very popular tool in the metrology industry because the experimental setup and acquisition phase with one imaging system is more straightforward than the ones using the stereo-vision technique. The only drawback in the experimental setup of the videogrammetry technique used in the industry is the requirement of targets / fiducial marks as points of interest which necessitate a lot of time to place. To ease the experimental setup, the fiducial marks can be projected on the object to measure but a problem still exists: it leads to sparse three-dimensional information contrary to dense 3D reconstruction using a speckle pattern textured object. Because our new methodology of calibration does not require fiducial marks anymore and using the DIC technique, it is now possible to measure the 3D shape of an object using the videogrammetry technique and without the need of targets: the calibration is realized with any quasi-planar speckle pattern textured object and if the natural texture of the object to reconstruct is not random enough, speckle pattern texture can be easily deposited using spray paint or speckle pattern image projection (see Fig. 3.9).

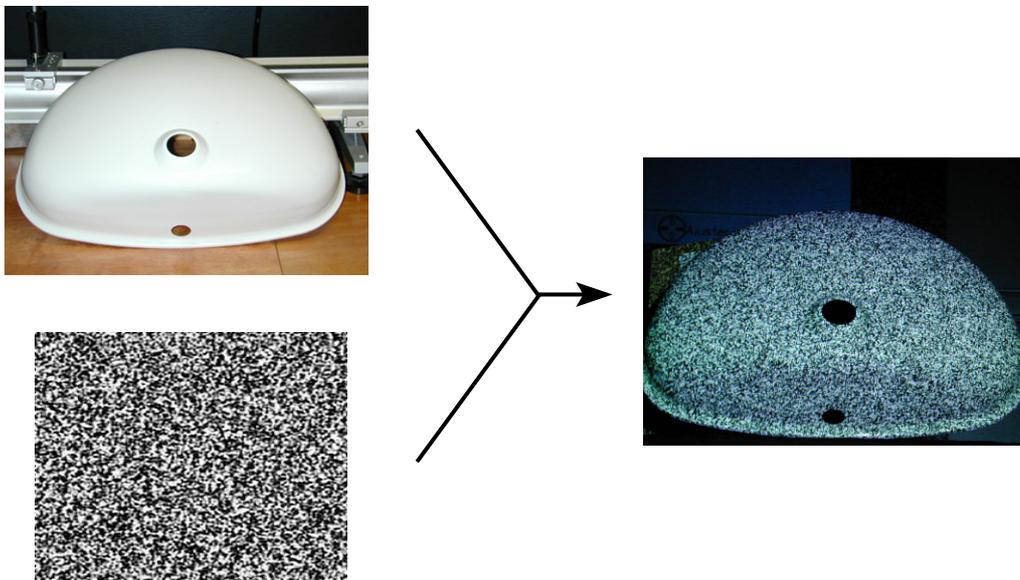


Fig. 3.9 — *Speckle pattern image projected onto an object to avoid to place target / fiducial marks.*

Applications of videogrammetry at different length-scales will be presented in the following sections: the 3D reconstruction of a penny detail at micro-scale [CGS⁺03] (see

Section 3.5.1) and the 3D reconstruction of a mold used for Pyrex[®] plate at macro-scale (see Section 3.5.2). For this last experiment, the 3D shape of the mold is also measured with a 3D laser scanner, giving us the opportunity to assess the accuracy of the videogrammetry technique and of the software we developed for the needs of the thesis (see Section 5).

3.5.1 3D Reconstruction of a Penny Detail

In order to validate on the one hand the spatially-varying distortion at low magnification (the drift being negligible at this range of magnification) and on the other hand the 3D reconstruction algorithm, an experiment has been conducted at a magnification of $200\times$ to measure the 3D shape of an American penny detail: the letter “R” of the sentence “E PLURIBUS UNUM”. Fig. 3.10 shows three fields of view for the penny. The $200\times$ coin detail shows a natural speckle pattern texture that is adequate for selective use of digital image correlation to identify and match corresponding subsets.

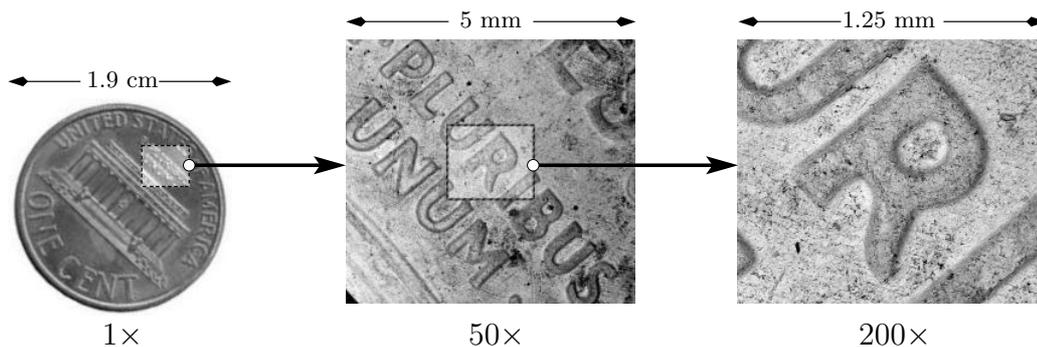


Fig. 3.10 — *Sample used for the 3D reconstruction: the letter “R” of the sentence “E PLURIBUS UNUM” of an American penny.*

For this experiment, the FEI ESEM Quanta 200 of the Electron Microscopy Center of the University of South Carolina is operated in high-vacuum mode (SEM mode). SEM operational parameters such as the detector type, accelerating voltage, working distance, *etc.* will affect the quality of the images. In this regard, the major requirements for image correlation in an SEM are (a) adequate image contrast, (b) random texture in the images, (c) appropriate spatial frequency in the random texture (d) temporal invariance in the images and (e) minimal image changes during the rigid-body motion of the specimen. Since SE detector imaging (which depends on topography of the sample) will violate (e) in most cases, BSE imaging is preferred. Moreover, the surface texture is generally improved when using the BSE detector. Since BSE imaging is also slightly affected by topography, all BSE images are acquired at a low accelerating voltage in this experiment. By this way, the primary beam electrons do not penetrate deeply into the specimen. Another advantage

of a low voltage incident beam is that surface details are enhanced in comparison to high voltage [RWT00]. The operating conditions for this study are then as follows:

- BSE detector
- 200× magnification
- non-dimensional spot-size of seven
- 10 mm working distance
- accelerating voltage of 8 kV
- seven-bit gray scale for all images (1024 × 884 size)
- 1.3 mm by 1.1 mm field of view

Calibration Procedure

The procedure used for this experiment is the simplified calibration procedure presented in Section 2.4. Two different calibration have been realized: a) an aluminum plate covered with a gold speckle pattern realized by a micro-lithography process (see Fig. 3.11) is employed as calibration target and b) the object to measure is directly employed as calibration target using the natural random texture on the American penny (see Fig. 3.10, right). The relative "flatness" of the coin detail (depth < 50 μ m) allows to ignore its real geometry and consider it planar as initial guess for the calibration.

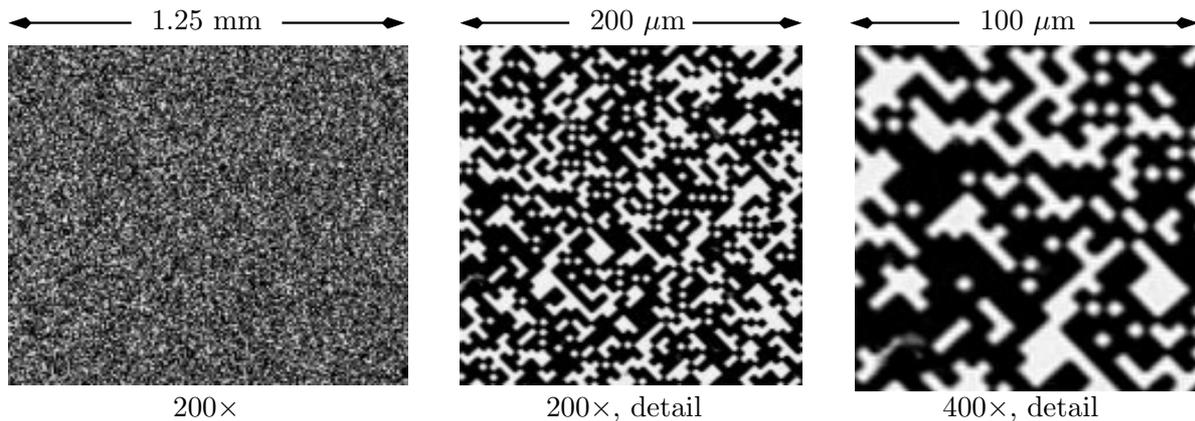


Fig. 3.11 — *BSE detector images of the planar aluminum calibration target covered with a thin speckle pattern layer of gold realized by micro-lithography.*

Experimental results using the gold speckle pattern realized by micro-lithography are not presented here but in the section about calibration: Section 2.4.3.

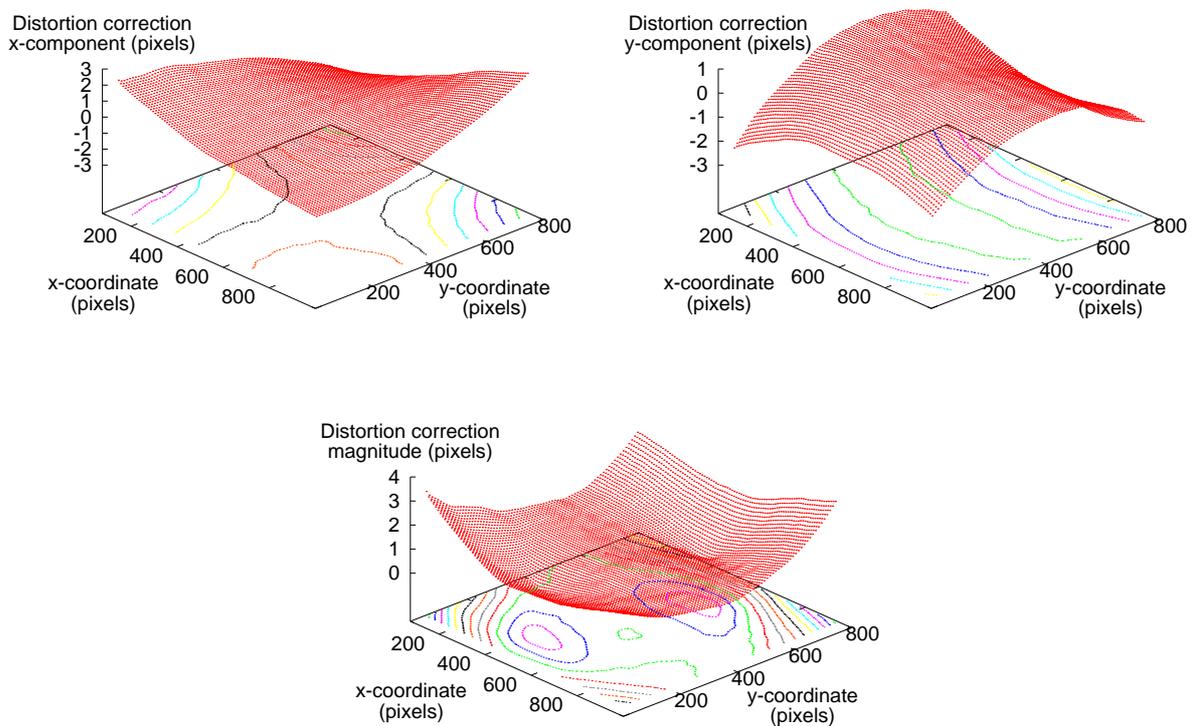


Fig. 3.12 — *Distortion correction field computed using the coin in replacement of the especially designed speckle pattern: x-component (left), y-component (right) and magnitude (bottom).*

Fig. 3.12 shows the distortion field of the SEM imaging system when using the random texture on the coin as calibration target. The results are very similar to those obtained by using the “ideal” planar target (Section 2.4.3).

Interestingly, the bundle adjustment technique reestimates the shape of the calibration target (the coin) while also computing the projection model parameters of the imaging system (see Fig. 2.21). Assuming a flat object as the initial guess for the calibration by bundle adjustment, Fig. 3.13 shows the final reestimated shape of the target.

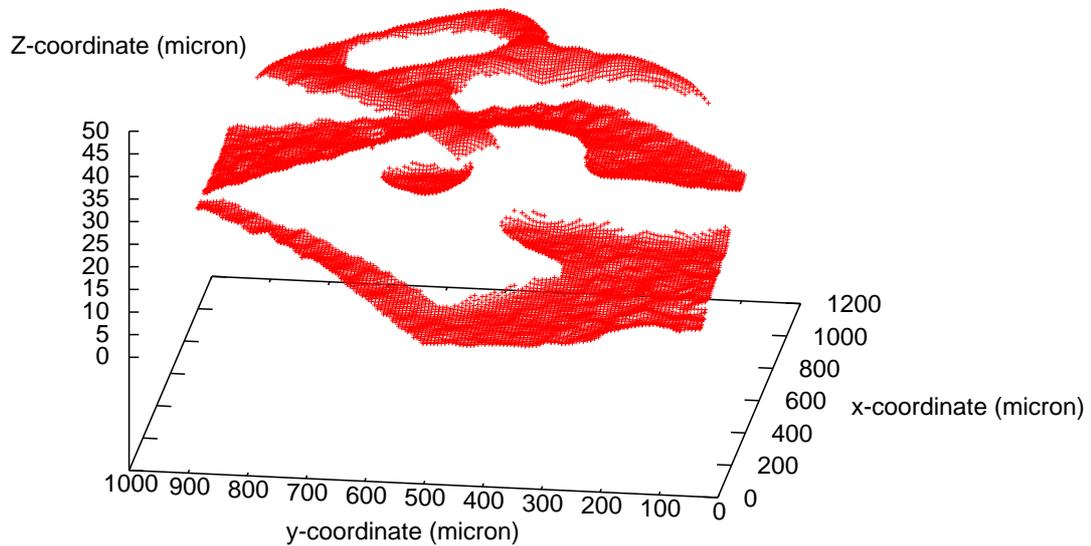


Fig. 3.13 — *Reestimated 3D shape of the coin target used for the calibration by bundle adjustment. The shape is made up of three distinct areas, which corresponds to the three areas of interest selected for being matched by correlation (we avoid to correlate areas of high curvature where the image correlation is likely to be less accurate).*

3D Reconstruction

The operating conditions for the acquisition of the images for the calibration and the ones for the 3D reconstruction must obviously be the same and we cannot remove the calibration target specimen from the SEM chamber after calibration to put the penny. The penny is then stuck near the gold speckle pattern to allow a new set of acquisitions after calibration without changing any parameters of the SEM (see Fig. 3.14).

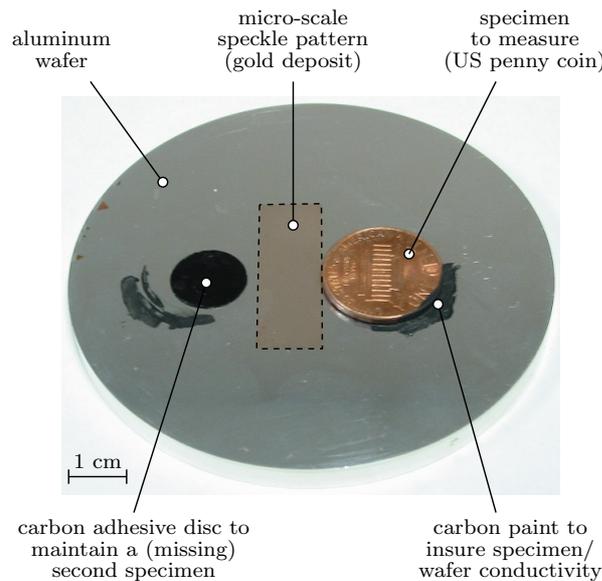


Fig. 3.14 — *Setup of the experiment: the coin to measure is stuck using a thin adhesive on an aluminum wafer covered with a gold speckle pattern deposited by micro-lithography.*

Even if two images would have been sufficient, for this experiment three images of the letter “R” of an American penny were acquired at $200\times$, under three different viewpoints (see Fig. 3.15).

Feature Points Extraction (see Fig. 3.16): A set of feature points is extracted for each image using Harris detector. This set is then processed to keep only the best feature points in a given circular neighborhood (typically a 5-pixel radius) with respect to their Harris “cornerness” function response. Using this approach, good feature points are regularly scattered throughout the image and the epipolar geometry to estimate in the next steps is likely to be better [Zha96a]. Depending upon the image being used, between 9200 and 9800 points are extracted.

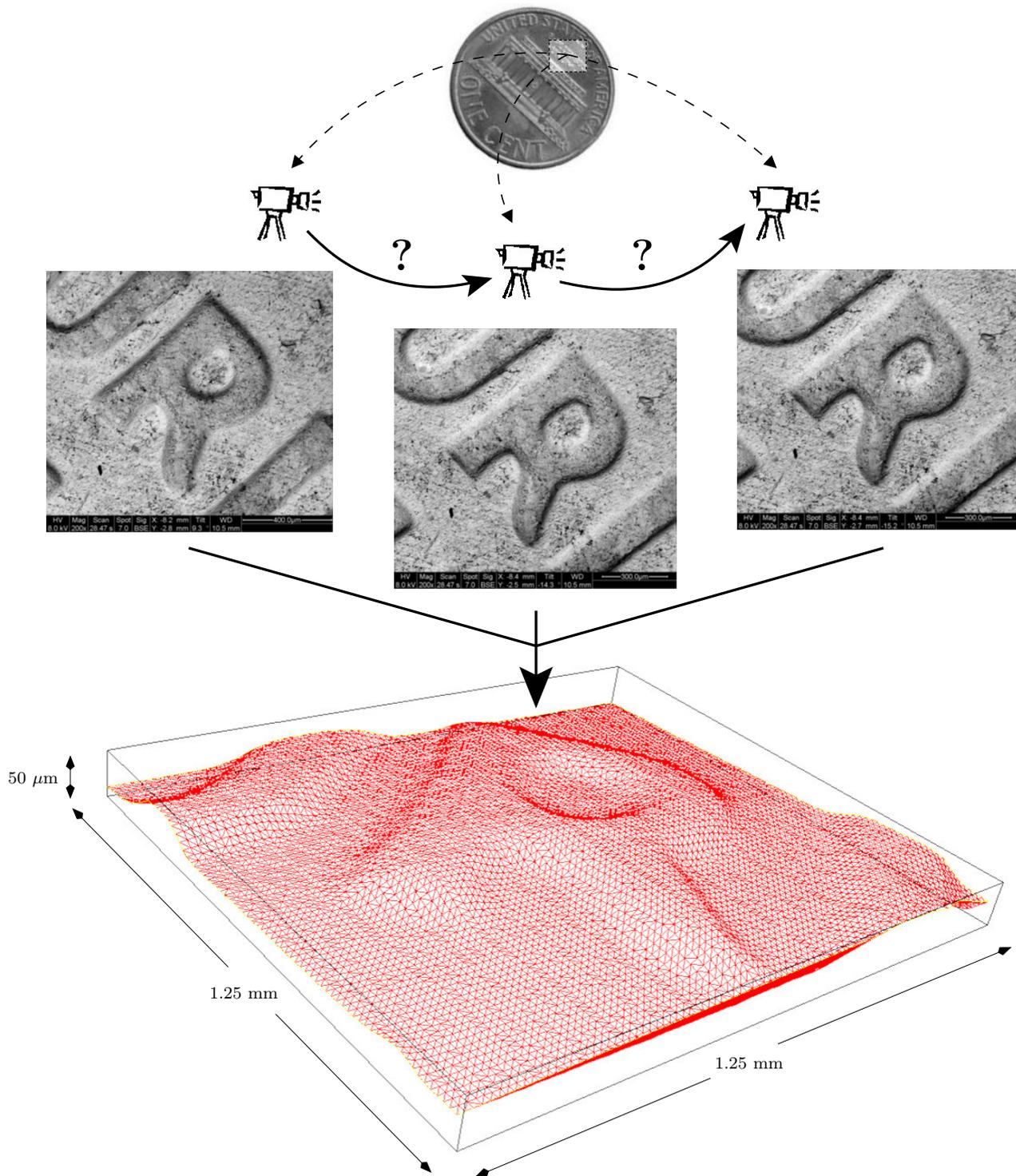


Fig. 3.15 — *Experiment of 3D reconstruction using videogrammetry: acquisition of the penny detail undergoing unknown rigid-body motions in order to obtain the 3D shape of the letter “R”.*

Robust Matching (see Fig. 3.16): For each pair of images (pair of images 1&2, pair of images 2&3 and pair of images 1&3), feature points are matched using:

- ZNCC criterion
- 15×15 pixels correlation window
- Correlation threshold³ of 70%
- Least Median of Squares method to detect and remove outliers (false matches) while estimating the epipolar geometry

An average of 3100 pairs of points are robustly matched from each set containing initially 9500 points (about 100 initial matches were removed as outliers).

Motion Estimation: As the feature points are only extracted with a pixel accuracy, the computation of the epipolar geometry is a first estimation for the following stage. At this point, the mean distance of points to their epipolar line is 0.26 pixels (see Equ. (3.7)). An estimation of the motion is recovered and a first approximation of the 3D shape is computed. Figure 3.17 shows the reconstructed shape based on 3100 matched feature points.

Dense Matching (see Fig. 3.18): When the epipolar geometry is estimated between each couple of images, the dense matching process is very simplified. Indeed, for each point in one image, its correspondence in the second image must lie on the associated epipolar line. Searching for the correspondence along the epipolar line⁴ is not only faster but also less error-prone because false matches are more easily avoided by this way. About 26000 points are correlated with a sub-pixel accuracy using DIC (ZNCC correlation criterion and affine transformation of the correlation window) for each pair of image. Epipolar geometry is finally refined (mean distance of points to their epipolar line is now 5.10^{-3} pixels) improving the motion estimates.

Triangulation / Bundle adjustment: Once the motions are accurately recovered, the triangulation process is achieved using the bundle adjustment technique. Figs. 3.19 and 3.20 show the 3D shape of the coin detail reconstructed by triangulation. The results in these figures confirm that the calibration and 3D reconstruction processes can be performed

³The ZNCC criterion returns a correlation score between 0 and 1, 1 corresponding to a perfect similarity measure. A correlation threshold of 70% means that the result of correlation is taken into account only if the ZNCC score is greater than 0.7.

⁴Actually, in practice, the correspondence is not only searched along the epipolar line but inside a very narrow band of a few pixels width defined around the line.

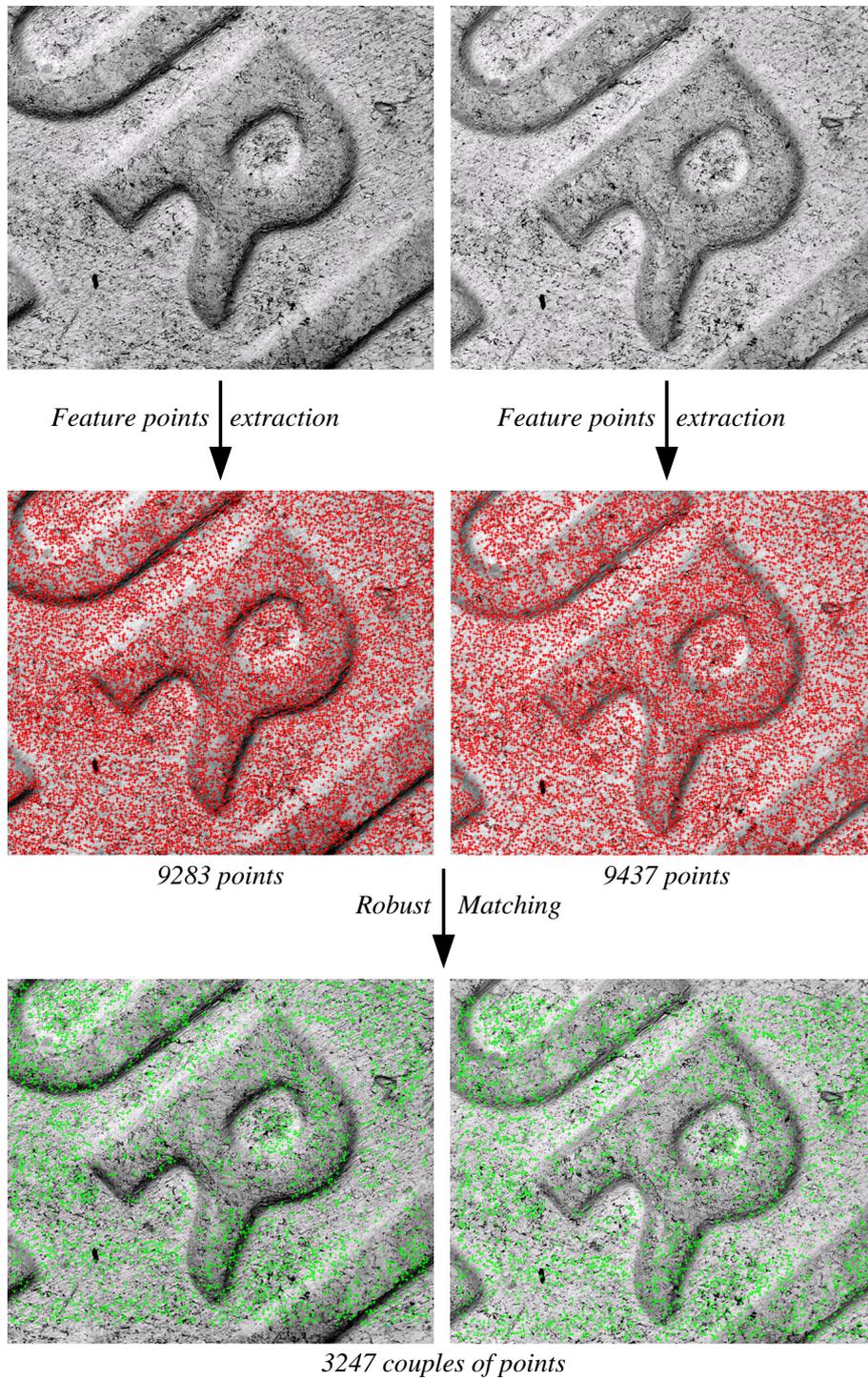


Fig. 3.16 — *Example of the first stages of the algorithm of reconstruction with two images: feature points extraction and robust matching.*

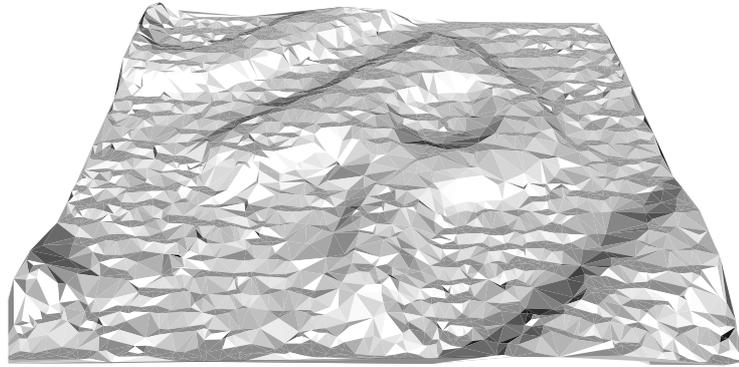
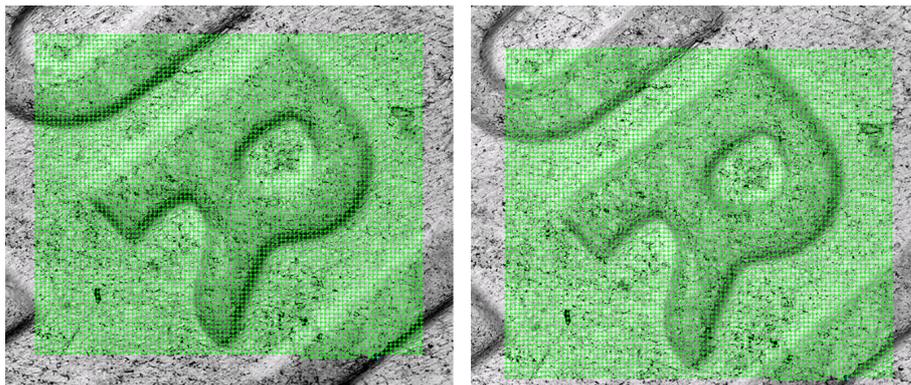


Fig. 3.17 — *Reconstructed 3D shape based upon the first estimation of the motions and using only the 3100 matched feature points: the reconstructed points are sparse and the quality of reconstruction is poor. Epipolar geometry, and consequently the motions, have to be refined.*



26400 couples of points

Fig. 3.18 — *Because the points correspondences are more numerous and correlated with a sub-pixel accuracy, the dense matching leads to a better estimation of the epipolar geometry and consequently of the motion.*

with arbitrary rigid-body motions using a single sensor imaging system such as the FEI ESEM.

The accuracy of the 3D reconstruction can hardly be assessed here because the real shape of the letter “R” of the penny is not available and it is difficult to determine by another method (contrary to macro-scale experiments where the 3D shape can be obtained using a 3D laser scanner for example: Section 3.5.2). However, the excellent results of calibration (the standard deviation of reprojection errors is 2.10^{-2} pixels) and the quality of the epipolar geometry estimation (mean distance of points to the epipolar line is 5.10^{-3} pixels) must lead to a very good estimation of the unknown motions and consequently of the 3D shape.

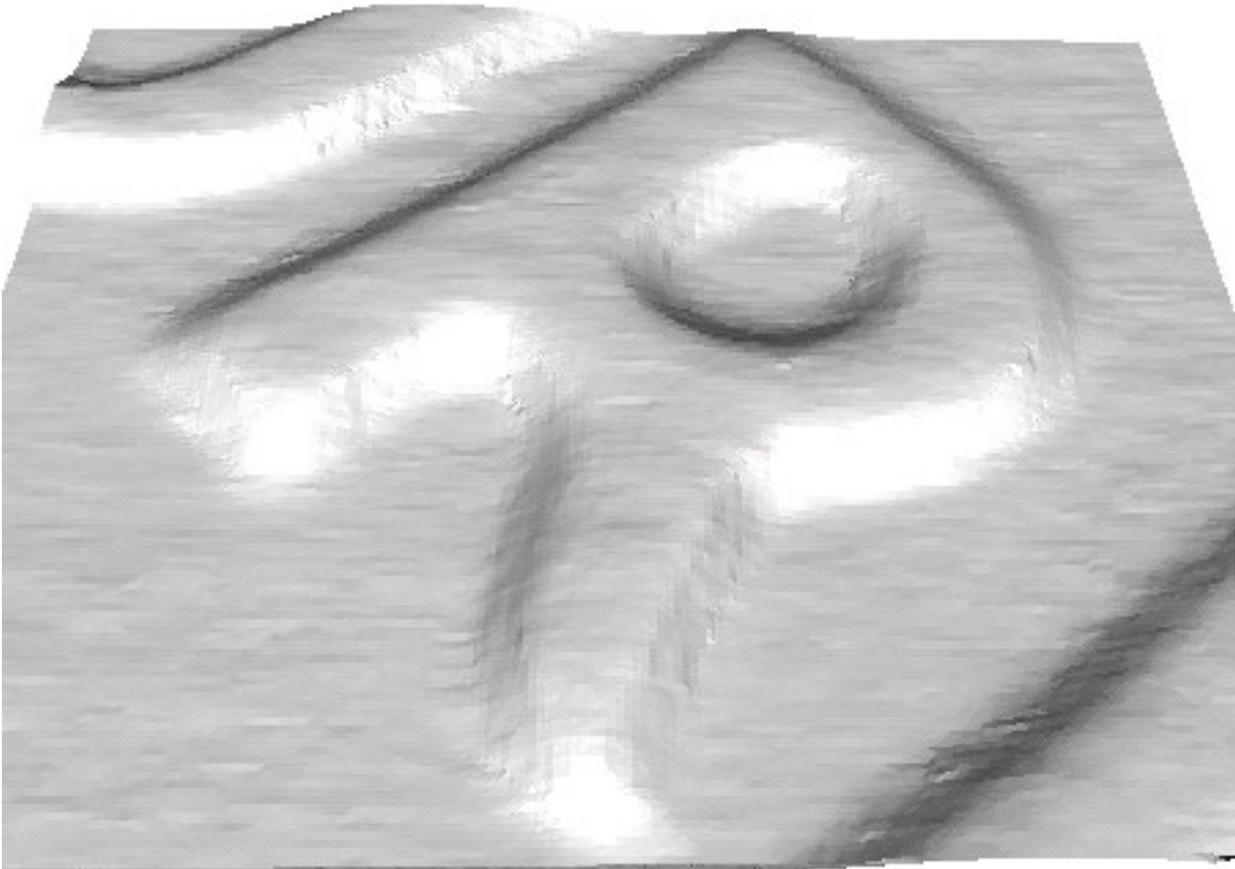


Fig. 3.19 — *Reconstructed 3D shape after dense sub-pixel matching by image correlation (rendering of 26400 sub-sampled 3D points).*



Fig. 3.20 — *Reconstructed 3D shape with texture mapping.*

3.5.2 3D Reconstruction of a Glass Mold

Experimental Context



Fig. 3.21 — *Upper part of a mold used for Pyrex[®] glass plate covered with a spray-painted speckle pattern.*

In order to assess the accuracy of the 3D shape measurement methodology and validate the software developed in this thesis (see Section 5), a comparative experiment has been conducted at the Research Centre on Tools, Materials and Forming Processes (CROMeP in French) of Ecole des Mines d'Albi-Carmaux. The centre, focused towards optimization of forming tools, was interested to test the videogrammetry technique for reconstructing the 3D shape of a part of a mold for Pyrex[®] glass plate (see Fig. 3.21) and to compare the results with the ones obtained using a commercial digitizing instrument, the Konica Minolta[®] VI-9i (see Fig. 3.22, right). This 3D laser scanner is used for reverse-engineering by the CIRTES, partner of Ecole des Mines and developer of the StratoConception[®] process [Bar92, BFGM95], a rapid prototyping technique able to realize a mold from a CAD model or a cloud of points. For the videogrammetry part, the experiment is carried out using a HAMAMATSU C4742-95 digital camera equipped with a 16 mm lens and mounted on a

tripod (see Fig. 3.22, left).



HAMAMATSU C4742-95 digital camera



Konica Minolta® VI-9i 3D digitizer

Fig. 3.22 — *The digital camera (left) is used for the 3D shape measurement of the glass mold using the videogrammetry technique. The 3D shape of the mold is measured as well using the 3D laser scanner (right) for comparison.*

3D Reconstruction using the Videogrammetry Technique

For the videogrammetry experiment, a calibration process is first necessary and eight images are acquired in different orientations and positions (see Fig. 3.23) of the camera. The calibration target is made with a speckle pattern printed and glued on a flat and rigid Plexiglas plate.

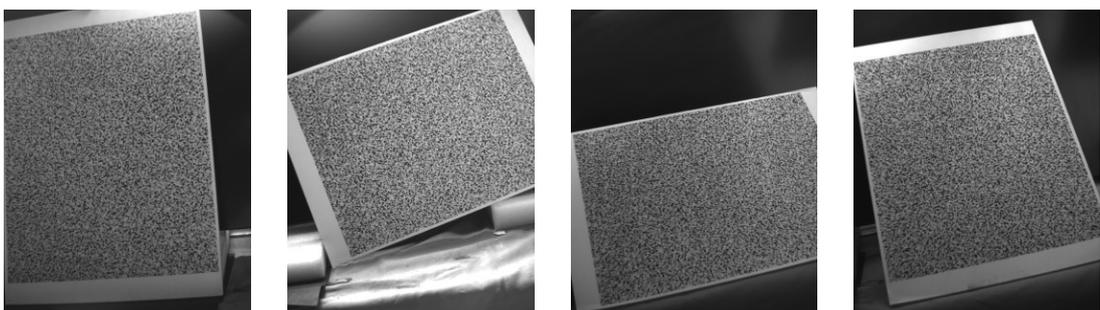


Fig. 3.23 — *Four images of the image sequence acquired for calibration.*

Calibration determines the projection model parameters (pinhole parameters in this case) of the camera and the spatially-varying distortion correction function (non-parametric and approximated by a bicubic spline of 10×10 patches). Indeed, there is no time-varying distortion for an imaging system such as a camera and the simplified calibration procedure presented in Section 2.4 is sufficient. Fig. 3.24 shows that the distortion is quite important here (from -4 to +3 pixels for the x-component and from -4 to +3 pixels for the y-component) which means that the 3D shape measurement without taking into account distortion would have been very inaccurate.

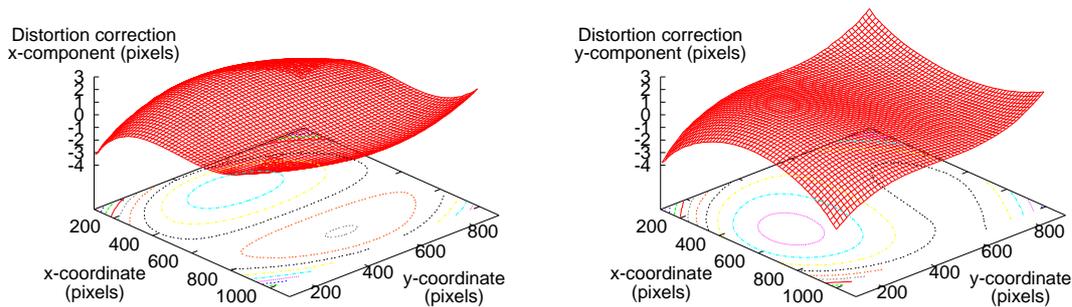


Fig. 3.24 — *Horizontal (left) and vertical (right) component of the distortion correction function: acquired images are very distorted (up to 4 pixels of distortion).*

For the 3D reconstruction, a speckle pattern texture using white spray-paint is realized on the mold and four images are acquired from different viewpoints (see Fig. 3.25). Because the distortion is important here, we preferred not using the feature point detector on undistorted images (to avoid loss of accuracy with the interpolation) and *Vic2D*[®] [CSI] is used for a dense correlation using the original images: an area of interest of more than 18000 points is selected in the first image and correlated in the image sequence. Coordinates of measure-points exported by *Vic2D*[®] are then corrected for distortion and our 3D reconstruction process using videogrammetry technique (see Section 3) is performed using the software we developed: *Visiocre* (see Section 5). The reconstructed 3D shape of the mold is presented in Fig. 3.26.

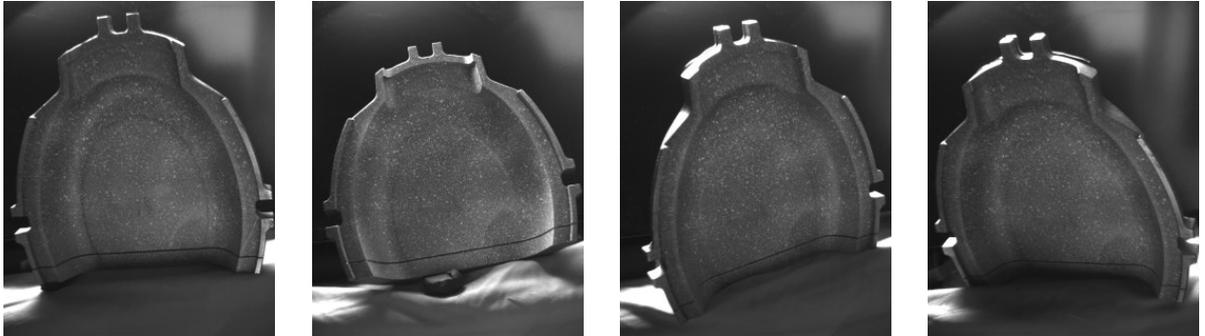


Fig. 3.25 — *The four images acquired by the HAMAMATSU digital camera and used for the 3D reconstruction using videogrammetry.*

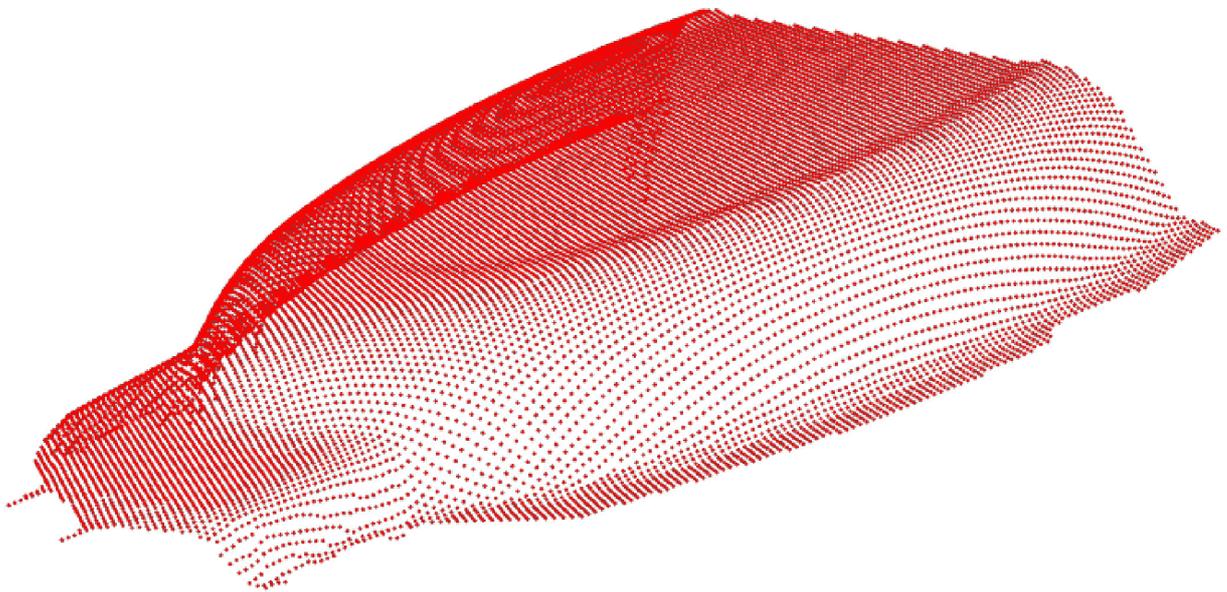


Fig. 3.26 — *3D shape of the glass mold (cloud of points) using the videogrammetry technique.*

3D Reconstruction using the 3D Laser Scanner

Using the 3D digitizer Konica Minolta[®] VI-9i, the mold is acquired in three steps (in front, left and right viewpoint) because of the curved parts inside the mold which cannot be seen from one single viewpoint. Three clouds of about 100000 points are then generated (see Fig. 3.27) and merged into a unique cloud by the software provided by the the laser scanning system. The final cloud contains a little more than 135000 points and is illustrated in Fig. 3.28.



Fig. 3.27 — *Three clouds of points generated by the 3D scanner, before the registration and fusion. We can notice that the curved parts of the mold are not reconstructed in the two upper clouds.*

Accuracy Assessment

The accuracy of the 3D shape measurement made using the videogrammetry technique will be assessed by comparing a cloud of points generated by our videogrammetry software (see Fig. 3.26) and the one generated by the 3D digitizer Konica Minolta[®] VI-9i (see Fig. 3.28) claiming a measurement accuracy of ± 0.05 mm. Because the 3D shapes are not in the same coordinate system, the comparison of the two shapes is realized using a program based on the ICP (Iterative Closest Point) algorithm and implemented by Andres Restrepo-Specht

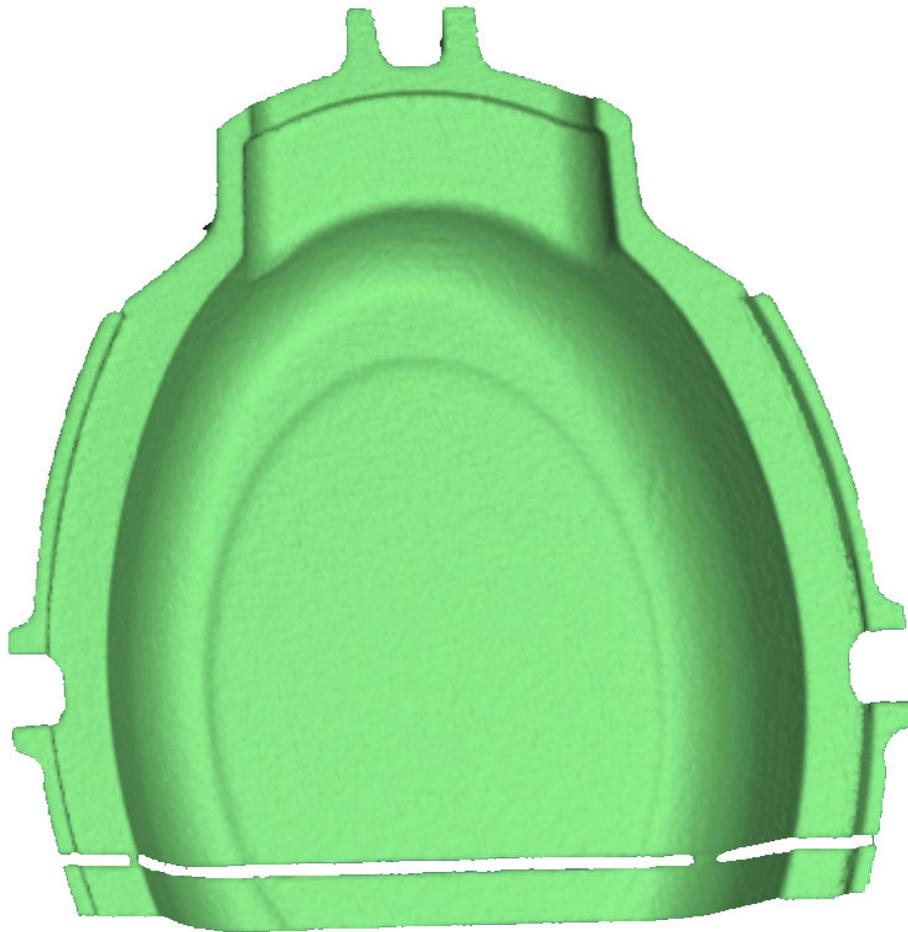


Fig. 3.28 — *Final cloud of more than 135000 points generated by the 3D laser scanner and used for the comparison with the cloud of points obtained with the videogrammetry technique (see Fig. 3.26).*

[RSDS02, RS05] during his PhD thesis carried out at LAAS-CNRS Toulouse⁵. This algorithm [Ben98, BS99] estimates iteratively the best rigid-body transformation to transform the first cloud of points into the second one (registration) minimizing the distance between each pair of corresponding points. The mean distance between all pairs will be used to assess the accuracy of the 3D reconstruction. With respect to the classical implementation, the approach here is modified in order to estimate a scale factor in addition to the rigid-body motion between the two clouds of points. Indeed, the 3D shape obtained by the videogrammetry technique is known only up to a scale factor. The optimization process is then composed of two steps: 1) estimation of the scale factor and of the rigid-body motion and (about 100 iterations) 2) refinement of the rigid-body motion considering the scale factor fixed (about 60 iterations). In the first step, the data are sub-sampled to converge faster (about only 6000 points per cloud are used) but in the second step, all the points are used. The cloud of points obtained from the 3D laser scanner being much more dense (about 130000 points) than the cloud from the videogrammetry technique (about 18000 points), each point of the second cloud has finally a correspondence in the first.

In order to show the importance of the distortion correction for the accuracy of the measurement, two 3D reconstruction are achieved using the videogrammetry technique: one from undistorted point correspondences as previously described denoted (a) and one from points directly obtained by the correlation and not corrected for distortion denoted (b). The two clouds look similar qualitatively (see Fig. 3.29) but a difference of accuracy nevertheless exists: the mean distance between the 3D shape obtained with the 3D laser scanner and (a) is 0.012 mm whereas the mean distance between the shape obtained with the 3D laser scanner and (b) is 0.028 mm. These results are very good and confirm that the videogrammetry technique and the software *Visiocre* developed for this work are as efficient as a 3D laser scanner. They prove as well that it is always necessary to take into account the spatially-varying distortion because it leads here to more than a 2-fold increase in accuracy.

⁵We wish to acknowledge Andres Restrepo-Specht and Michel Devy from LAAS-CNRS Toulouse for their generous and constant help in this experiment.

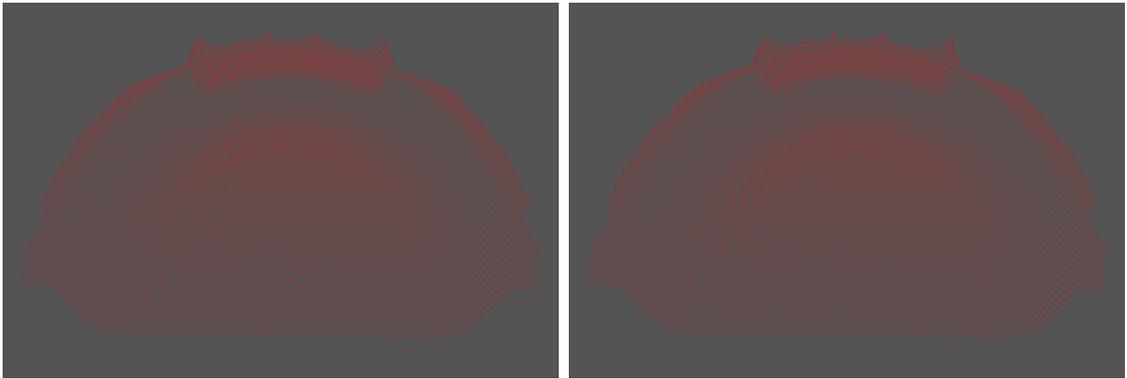


Fig. 3.29 — *Qualitative comparison between the 3D shapes with (left) and without distortion correction (right): at the naked-eye, the 3D shapes look similar but a difference in accuracy measurement exists.*

Bibliography

- [AS03] X. Armangue and J. Salvi. Overall View Regarding Fundamental Matrix Estimation. *Image and Vision Computing*, 21(2):205–220, 2003.
- [Bar92] C. Barlier. Le procédé de prototypage rapide par STRATOCONCEPTION®. In *Proceedings of the first "Assises Européennes du Prototypage Rapide"*, Palaiseau (France), Jun 1992.
- [Ben98] Raouf Benjemaa. *Traitement de données 3D denses acquises sur des objets réels complexes*. PhD thesis, ENST, Paris, Oct 1998.
- [BFGM95] C. Barlier, U. Feltes, D. Gasser, and F. Muller. STRATOCONCEPTION®, Rapid Prototyping for die-forging tooling. In *Proceedings of 28th International Symposium on Automotive Technology and Automation (ISATA)*, Stuttgart (Germany), Sep 1995.
- [BS99] R. Benjemaa and F. Schmitt. Fast global registration of 3D sampled surfaces using a multi-z-buffer technique. *Image and Vision Computing*, 17(2):113–123, 1999.
- [Can86] John F. Canny. A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–714, 1986.
- [CGS⁺03] Nicolas Cornille, Dorian Garcia, Michael A. Sutton, Stephen R. McNeill, and Jean-José Orteu. Automated 3D Reconstruction Using a Scanning Electron Microscope. In *SEM Conference on Experimental and Applied Mechanics*, Charlotte (NC, USA), Jun 2003.
- [CGS⁺04] Nicolas Cornille, Dorian Garcia, Michael A. Sutton, Stephen R. McNeill, and Jean-José Orteu. Calibrage d'un MEB en vue de la mesure précise de formes et de déformations 3D. In *Proceedings of Colloque Photomécanique 2004*, Albi (France), 2004.
- [CLVWdV02] E. Cecchi, J.M. Lavest, and B. Van Wyk de Vries. Videogrammetric reconstruction applied to volcanology: perspectives for a new measurement technique in volcano monitoring. *Image Analysis and Stereology*, 21(1):31–36, 2002.
- [Cot94] J.C. Cottier. Extraction et appariements robustes des points d'intérêts de deux images non étalonnées. Technical report, LIFIA-IMAG-INRIA, 1994.

-
- [CSG⁺04] Nicolas Cornille, Hubert W. Schreier, Dorian Garcia, Michael A. Sutton, Stephen R. McNeill, and Jean-José Orteu. Gridless Calibration of Distorted Visual Sensors. In *Proceedings of Colloque Photomécanique 2004*, Albi (France), 2004.
- [CSI] Vic2D[®] and Vic3D[®], Correlated Solutions Inc., <http://www.correlatedsolutions.com>.
- [Fau93] Olivier Faugeras. *Three-Dimensional Computer Vision: A Geometric Viewpoint*. MIT Press, 1993. ISBN 0-262-06158-9.
- [Fau95] Olivier Faugeras. Stratification of 3D Vision: Projective, Affine and Metric Representations. *Journal of the Optical Society of America A*, 12(3):465–484, Mar 1995.
- [FLP01] O. Faugeras, Q.-T. Luong, and T. Papadopolou. *The Geometry of Multiple Images: The Laws That Govern The Formation of Images of A Scene and Some of Their Applications*. MIT Press, 2001.
- [FM95] O. D. Faugeras and B. Mourrain. On the geometry and algebra of the point and line correspondences between n images. In *Proceedings of the 5th International Conference on Computer Vision (ICCV'95)*, pages 951–962, Boston (MA, USA), 1995.
- [For94] Wolfgang Forstner. A framework for low-level feature extraction. In *Proceedings of the 3rd European Conference on Computer Vision (ECCV'94)*, pages 383–394, Stockholm (Sweden), 1994.
- [Har95] Richard Hartley. In Defence of the 8-point Algorithm. In *5th International Conference on Computer Vision (ICCV'95)*, pages 1064–1070, Boston (MA, USA), Jun 1995.
- [Har98] Richard Hartley. Computation of the Quadrifocal Tensor. In *5th European Conference on Computer Vision (ECCV'98)*, volume 1406, pages 20–35, Freiburg (Germany), Jun 1998.
- [HRVDH⁺92] F. Heitger, L. Rosenthaler, R. Von Der Heydt, E. Peterhans, and O. Kuebler. Simulation of neural contour mechanism: from simple to end-stopped cells. *Vision Research*, 32(5):963–981, 1992.
- [HS88] C. Harris and M. Stephens. A Combined Corner and Edge Detector. In *Alvey Vision Conference*, pages 147–151, 1988.
-

- [HS97] M. Hemmleb and M. Schubert. Digital microphotogrammetry - Determination of the topography of microstructures by scanning electron microscope. In *Second Turkish-German Joint Geodetic Days*, pages 745–752, Berlin (Germany), May 1997.
- [HSV90] R. Horaud, T. Skordas, and F. Veillon. Finding geometric and relational structures in an image. In *Proceedings of the 1st European Conference on Computer Vision (ECCV'90)*, pages 374–384, Antibes (France), 1990.
- [HZ00] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.
- [JJ95] Drahomira Janova and Jiri Jan. Robust Surface Reconstruction from Stereo SEM Images. In V. Hlavac and R. Sara, editors, *Computer Analysis of Images and Patterns (CAIP'95)*, pages 900–905, Berlin (Germany), 1995. Springer.
- [JL01] I.K. Jung and S. Lacroix. A robust interest points matching algorithm. In *8th International Conference on Computer Vision (ICCV'2001)*, Vancouver (Canada), Jul 2001.
- [Lav96] Stéphane Laveau. *Géométrie d'un système de n caméras. Théorie, estimation et applications*. PhD thesis, Ecole Polytechnique, May 1996.
- [LBVP⁺04] Guy Le Besnerais, Fabien Vignon, Jean-Louis Pouchou, Denis Boivin, and Pierre Beauchene. Surface Reconstruction from Multiple SEM Images. In *Workshop on Applications of Computer Vision 2004*, Orlando (FL, USA), Dec 2004.
- [LF96] Q.-T. Luong and O. Faugeras. The fundamental matrix: Theory, algorithms, and stability analysis. *International Journal of Computer Vision*, 17(1):43–76, 1996.
- [LH81] Hugh Christopher Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135, Sep 1981.
- [LTCY98] A.J. Lacey, S. Thacker, S. Crossley, and R.B. Yates. A multi-stage approach to the dense estimation of disparity from stereo SEM images. *Image and Vision Computing*, 16:373–383, 1998.
- [MeX] MeX[®] software, Alicona Imaging, <http://www.alicon.com/>.
- [MZ92] J.-L. Mundy and A. Zisserman. *Geometric Invariance in Computer Vision*. MIT Press, 1992.

-
- [Pen55] R.A. Penrose. A Generalized Inverse for Matrices. In *Proceedings of the Cambridge Philosophical Society*, volume 51, pages 406–413, 1955.
- [RL87] P. Rousseeuw and A. Leriy. *Robust Regression and Outlier Detection*. John Wiley & Sons, 1987.
- [RS05] A. Restrepo-Specht. *Modélisation d’objets 3D par construction incrémentale d’un maillage triangulaire, dans un contexte robotique*. PhD thesis, Université Paul Sabatier, Jan 2005.
- [RSDS02] A. Restrepo-Specht, M. Devy, and A. Sappa. Robustness to initial conditions in registration of range images using the ICP algorithm. In *Proceedings of 3rd International Symposium on Robotics and Automation, ISRA’2002*, Toluca (Mexico), Sep 2002.
- [RWT00] R.G. Richards, M. Wieland, and M. Textor. Advantages of stereo imaging of metallic surfaces with low voltage backscattered electrons in a field emission scanning electron microscope. *Journal of Microscopy*, 199:115–123, Aug 2000.
- [SA91] M. E. Spetakis and J. Aloimonos. A multi-frame approach to visual motion perception. *International Journal of Computer Vision*, 16(3):245–255, 1991.
- [SAM] SAMx[®], 3D_TOPx package, <http://www.samx.com>.
- [Sas94] A. Sashua. Trilinearity in visual recognition by alignment. In *Proceedings of the 3rd European Conference on Computer Vision (ECCV’94)*, volume 1, pages 479–484, Stockholm (Sweden), May 1994.
- [SB97] S. M. Smith and J. M. Brady. SUSAN - a new approach to low-level image processing. *International Journal of Computer Vision*, 23(1):45–78, 1997.
- [SMB00] C. Schmid, R. Mohr, and C. Bauckhage. Evaluation of Interest Point Detectors. *International Journal of Computer Vision*, 37(2):151–172, 2000.
- [TH84] R. Y. Tsai and T. S. Huang. Uniqueness and estimation of three-dimensional motion parameters of rigid objects with curved surfaces. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 6:13–26, 1984.
- [TK92] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: a factorization method. *International Journal of Computer Vision (IJCV)*, 9(2):137–154, 1992.
-

- [TZ97] P. H. S. Torr and A. Zisserman. Robust Parameterization and Computation of the Trifocal Tensor. *Image and Vision Computing*, 15(8):591–605, Aug 1997.
- [VL93] T. Viéville and Q. Luong. Motion of points and lines in the uncalibrated case. Technical Report RR-2054, INRIA, 1993.
- [VL01] E. Vincent and E. Laganière. Matching Features Points in Stereo Pairs: A Comparative Study of Some Matching Strategies. *Machine Graphics & Vision*, 10(3):237–259, 2001.
- [VLBB⁺01] F. Vignon, G. Le Besnerais, D. Boivin, J.-L. Pouchou, and L. Quan. 3D reconstruction from scanning electron microscopy using stereovision and self-calibration. In *Physics in Signal and Image Processing*, Marseille (France), Jan 2001.
- [Zha96a] Zhengyou Zhang. A New Multistage Approach to Motion and Structure Estimation: From Essential Parameters to Euclidean Motion via Fundamental Matrix. Technical Report RR-2910, INRIA, Jun 1996.
- [Zha96b] Zhengyou Zhang. Determining the Epipolar Geometry and its Uncertainty: A Review. Technical Report RR-2927, INRIA, Jul 1996.

Chapter 4

Surface Strain Measurement

Similarly to the increasing interest for 3D shape measurement at micro- and nano-scale, development of nano-technology creates as well a need for strain measurement at these reduced-length scales. Digital image correlation is still the non-contacting method we preferred for strain measurement applications. Actually, deformation cannot be directly measured using the DIC technique: comparison of images of two (or more) different loads allows to measure only displacements, which are then used to compute strains. Because we do not observe under the surface, we will only treat the surface deformations in our studies and will not pay attention to the volumetric deformations [BSFS99, BDM04].

Section 4.1 introduces the methodology for computing in-plane strain¹ from the 2D displacement field. It presents as well the limitations in strain measurement due to the used imaging system: the Scanning Electron Microscope. Section 4.2 deals with three-dimensional strain² and the way to obtain them from the two reconstructed 3D shapes at two different times. It proposes as well methods to address the issues raised by the fact that SEM provides only one sensor.

Experimental results are presented in Section 4.3. In order to validate the entire process of calibration, 3D reconstruction and strain computation, a “zero-deformation” experiment is conducted: several images of a specimen undergoing only rigid-body motions are acquired and strains (supposed to be null) are computed between the two 3D shapes reconstructed by videogrammetry.

¹We call “in-plane strain” or “two-dimensional strain” the surface strain computed on a planar object.

²We call “three-dimensional strain” the surface strain computed on a 3D object.

4.1 In-Plane Strain Measurement

In-plane strain measurement³ at macro-scale using digital cameras and DIC technique began 20 years ago [MNSW⁺83, SWP⁺83, CRSP85, SCP⁺86] and is very common nowadays in experimental mechanics. The methodology is very simple: after a calibration phase of the imaging sensor, images corresponding to different states of deformation of the studied specimen are acquired⁴ (two-dimensional strain measurement only requires one image per state). Then, the area of interest of the object in the undeformed state, represented by a set of points regularly distributed or not, is compared by correlation to every other image in the sequence to obtain disparity maps (or displacement fields). Note that it is better to compare a deformed image with respect to the initial image instead of comparing it with a previous image in the sequence in order to avoid to propagate and cumulate the correlation errors. However, this is sometimes difficult in practice (in the case of large deformation for example) and a sequential approach is then employed. Finally, the strains are obtained by derivation of the displacement fields (see Section 4.1.1).

Obviously, in-plane strain measurement at micro- or nano-scale follows exactly the same procedure [Dou00]. The only issue comes from the used imaging system: because acquisition of an image with a SEM can last from some seconds to a few minutes, the deformation must absolutely not be significant during this period of time so that the object remains constant. Using a SEM imposes then a limitation on the rate of deformation: fast deformation can definitely not be measured using such an imaging system.

4.1.1 Strain Tensor Computation

Consider a set of original 2D points $\mathbf{m}_i = (x_i \ y_i)^\top$ which is transformed in the set of points $\mathbf{m}'_i = (x'_i \ y'_i)^\top$ (see Fig. 4.1). Coordinates x_i and y_i are called material coordinates and coordinates x'_i and y'_i are called spatial coordinates. They are related by the functions \mathcal{U} and \mathcal{V} representing respectively the value of the displacement in horizontal and vertical directions for a given point and defined such that:

$$\forall i, \quad \underbrace{x'_i}_{\text{deformed}} = \underbrace{x_i}_{\text{initial}} + \underbrace{\mathcal{U}(x_i, y_i)}_{\text{displacement}}$$

$$\underbrace{y'_i}_{\text{deformed}} = \underbrace{y_i}_{\text{initial}} + \underbrace{\mathcal{V}(x_i, y_i)}_{\text{displacement}}$$

Actually, it is oftentimes more convenient to use the functions \mathcal{X}' and \mathcal{Y}' (later referred to as “displacement functions”) defined by:

³What we call for convenience “strain measurement” is not really a direct measurement but actually a strain computation from the displacement measurement.

⁴Particular attention must be paid to the position of the camera: if the surface of the object is not fronto-parallel to the imaging sensor, a parasite perspective distortion will be included in the strain field.

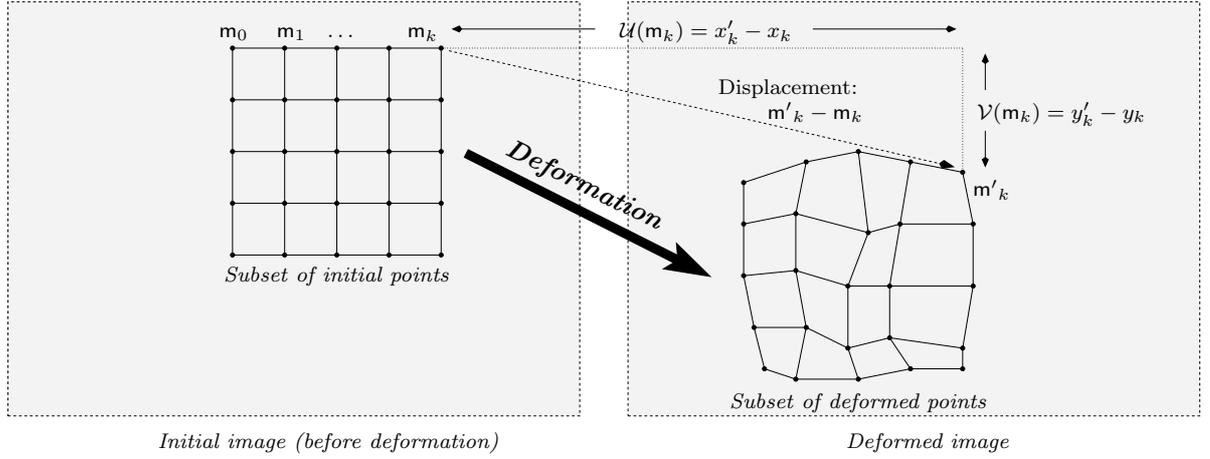


Fig. 4.1 — *In-plane strain: the set of 2D points \mathbf{m}_i are deformed into the set of 2D points \mathbf{m}'_i .*

$$\begin{aligned} \forall i, \mathcal{X}'(x_i, y_i) &= x'_i \\ \mathcal{Y}'(x_i, y_i) &= y'_i \end{aligned}$$

In order to derive the strains from the displacements, we need to introduce the deformation gradient tensor \mathbf{F} . Let's consider a differential vector $\vec{\mathbf{d}}_i$ of origin x_i that is transformed in the vector $\vec{\mathbf{d}}'_i$ of origin x'_i (see Fig. 4.2). \mathbf{F} is defined such that:

$$\vec{\mathbf{d}}'_i = \mathbf{F} \vec{\mathbf{d}}_i \quad (4.1)$$

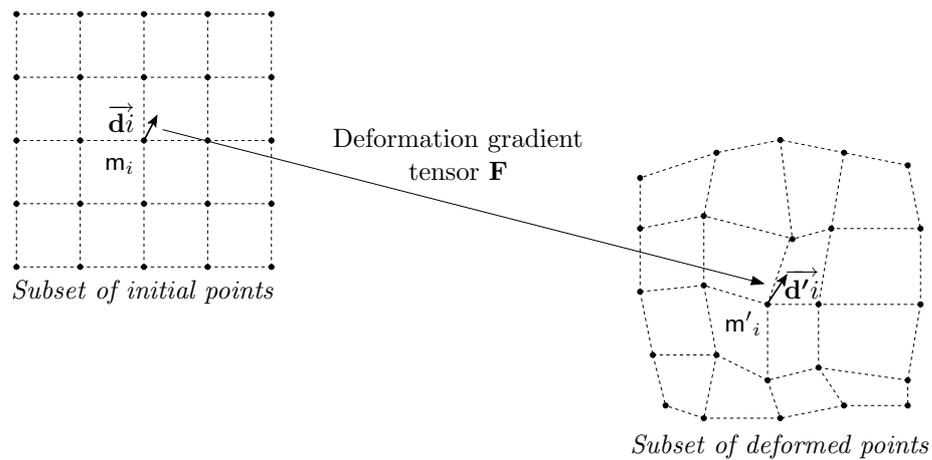


Fig. 4.2 — *The deformation gradient tensor \mathbf{F} defines the relationship between a vector $\vec{\mathbf{d}}_i$ of the initial surface and a vector $\vec{\mathbf{d}}'_i$ of the deformed surface.*

The tensor is then represented as follows:

$$\begin{aligned}
 \mathbf{F} &= \begin{pmatrix} \frac{\partial \mathcal{X}'}{\partial x} & \frac{\partial \mathcal{X}'}{\partial y} \\ \frac{\partial \mathcal{Y}'}{\partial x} & \frac{\partial \mathcal{Y}'}{\partial y} \end{pmatrix} \\
 &= \begin{pmatrix} 1 + \frac{\partial \mathcal{U}}{\partial x} & \frac{\partial \mathcal{U}}{\partial y} \\ \frac{\partial \mathcal{V}}{\partial x} & 1 + \frac{\partial \mathcal{V}}{\partial y} \end{pmatrix}
 \end{aligned} \tag{4.2}$$

The dilatation tensor or Cauchy-Green tensor \mathbf{C} is defined to characterize the dilatation of the original vector $\vec{\mathbf{d}}_i$:

$$\begin{aligned}
 \|\vec{\mathbf{d}}'_i\|^2 &= \vec{\mathbf{d}}'_i{}^\top \cdot \vec{\mathbf{d}}'_i \\
 &= \vec{\mathbf{d}}_i{}^\top \mathbf{F}^\top \cdot \mathbf{F} \vec{\mathbf{d}}_i \quad \text{using Equ. (4.1)} \\
 &= \vec{\mathbf{d}}_i{}^\top \mathbf{C} \vec{\mathbf{d}}_i \\
 \text{where } \mathbf{C} &= \mathbf{F}^\top \mathbf{F}
 \end{aligned} \tag{4.3}$$

Finally, the Green-Lagrange strain tensor⁵ \mathbf{E} is determined from the Cauchy-Green tensor \mathbf{C} :

$$\mathbf{E} = \frac{1}{2} (\mathbf{C} - \mathbf{I}) \quad \text{where } \mathbf{I} \text{ is the identity matrix}$$

Note that \mathbf{E} is symmetric because \mathbf{C} is symmetric according to Equ. (4.3). \mathbf{E} is represented as follows:

$$\mathbf{E} = \begin{pmatrix} E_{xx} & E_{xy} \\ E_{xy} & E_{yy} \end{pmatrix}$$

where E_{xx} , E_{yy} are the normal strains and E_{xy} is the shear strain. Using Equ. (4.2) and Equ. (4.3), they can be identified to:

$$\begin{aligned}
 E_{xx} &= \frac{1}{2} \left[\left(\frac{\partial \mathcal{X}'}{\partial x} \right)^2 + \left(\frac{\partial \mathcal{Y}'}{\partial x} \right)^2 - 1 \right] \\
 E_{yy} &= \frac{1}{2} \left[\left(\frac{\partial \mathcal{X}'}{\partial y} \right)^2 + \left(\frac{\partial \mathcal{Y}'}{\partial y} \right)^2 - 1 \right] \\
 E_{xy} &= \frac{1}{2} \left(\frac{\partial \mathcal{X}'}{\partial x} \frac{\partial \mathcal{X}'}{\partial y} + \frac{\partial \mathcal{Y}'}{\partial x} \frac{\partial \mathcal{Y}'}{\partial y} \right)
 \end{aligned}$$

⁵Note that another formulation of the strain tensor exists: the Euler-Almansi strain tensor. Instead of expressing the strain tensor as a function of the material coordinates x_i and y_i (point of the undeformed surface), the Euler-Almansi tensor is expressed as a function of the spatial coordinates x'_i and y'_i (point of the deformed surface).

Replacing \mathcal{X}' by $(\mathcal{U} + x)$ and \mathcal{Y}' by $(\mathcal{V} + y)$ leads to expressions of E_{xx} , E_{yy} and E_{xy} depending on \mathcal{U} and \mathcal{V} :

$$\begin{aligned} E_{xx} &= \frac{\partial \mathcal{U}}{\partial x} + \frac{1}{2} \left[\left(\frac{\partial \mathcal{U}}{\partial x} \right)^2 + \left(\frac{\partial \mathcal{V}}{\partial x} \right)^2 \right] \\ E_{yy} &= \frac{\partial \mathcal{V}}{\partial y} + \frac{1}{2} \left[\left(\frac{\partial \mathcal{U}}{\partial y} \right)^2 + \left(\frac{\partial \mathcal{V}}{\partial y} \right)^2 \right] \\ E_{xy} &= \frac{1}{2} \left(\frac{\partial \mathcal{U}}{\partial y} + \frac{\partial \mathcal{V}}{\partial x} + \frac{\partial \mathcal{U}}{\partial x} \frac{\partial \mathcal{U}}{\partial y} + \frac{\partial \mathcal{V}}{\partial x} \frac{\partial \mathcal{V}}{\partial y} \right) \end{aligned}$$

Strains⁶ can then be determined from the partial derivatives of the displacement functions \mathcal{X}' and \mathcal{Y}' or \mathcal{U} and \mathcal{V} . These functions are sampled and measured at discrete number of locations. Therefore, they are approximated by using the sets of initial and deformed 2D points and by fitting these data with a transformation such as a local affine transformation, a local quadratic transformation or a global spline function. Only the approximation of the functions \mathcal{X}' and \mathcal{Y}' will be presented in Section 4.1.2 because the approximation \mathcal{U} and \mathcal{V} is similar and does not need to be detailed.

4.1.2 Displacement Functions Approximation

The displacement functions can be approximated by transformations that will be presented thereafter. The affine transformation is the simplest representation and can be considered sufficient if the deformation in the region of interest is homogeneous (*i.e.*, constant extensional and shear strain). Quadratic and/or spline transformations are used if the deformations are non-homogeneous in the region of interest. Due to the presence of noise in the displacement measurements, the selection of an optimal displacement function necessitates the use of smoothing to reduce errors in the resulting displacement gradients.

Local approximation of the displacement functions by an affine transformation: Consider a subset of n points \mathbf{m}_k around the point \mathbf{m} where we want to compute the strains. These points are chosen inside a circle of a given radius or inside a square of a

⁶Note that in the case of small deformation, the Green-Lagrange strain tensor is generally approximated by its linearized counterpart $\boldsymbol{\varepsilon} = \frac{1}{2}(\mathbf{F}^\top + \mathbf{F}) - \mathbf{I}$.

$$\boldsymbol{\varepsilon} = \begin{pmatrix} \varepsilon_{xx} & \varepsilon_{xy} \\ \varepsilon_{xy} & \varepsilon_{yy} \end{pmatrix} \quad \text{where} \quad \begin{cases} \varepsilon_{xx} &= \frac{\partial \mathcal{X}'}{\partial x} - 1 = \frac{\partial \mathcal{U}}{\partial x} \\ \varepsilon_{yy} &= \frac{\partial \mathcal{Y}'}{\partial y} - 1 = \frac{\partial \mathcal{V}}{\partial y} \\ \varepsilon_{xy} &= \frac{1}{2} \left(\frac{\partial \mathcal{X}'}{\partial y} + \frac{\partial \mathcal{Y}'}{\partial x} \right) = \frac{1}{2} \left(\frac{\partial \mathcal{U}}{\partial y} + \frac{\partial \mathcal{V}}{\partial x} \right) \end{cases}$$

However, in order to stay in the most general case, only the strain tensor \mathbf{E} is considered in the following sections.

given side length (see Fig. 4.3). Some authors [ABBC94, Dou00] prefer to use a triangular mesh and the points are chosen inside a triangle, a square or an hexagon.

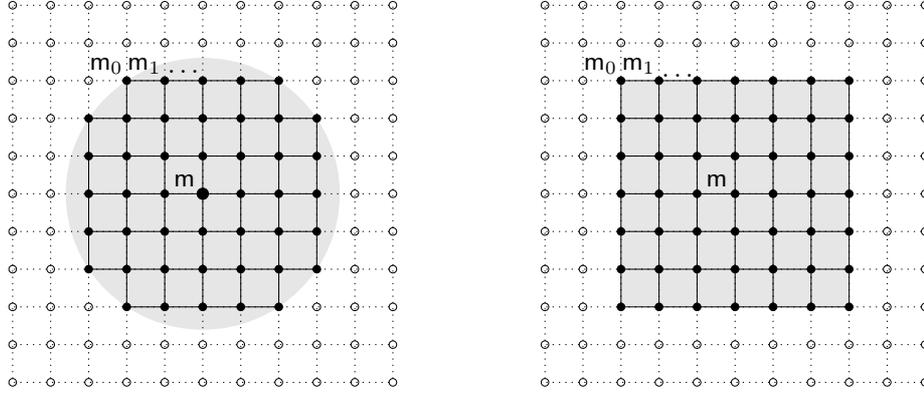


Fig. 4.3 — *Two types of possible neighborhood around the interest point m where the displacement functions will be approximated.*

The displacement functions between m_k and m'_k can be expressed locally as an affine transformation over the area defined by the point subset:

$$\begin{aligned} \forall k = 1 \dots n, \begin{pmatrix} x'_k \\ y'_k \end{pmatrix} &= \begin{pmatrix} \mathcal{X}'(x_k, y_k) \\ \mathcal{Y}'(x_k, y_k) \end{pmatrix} \\ &\approx \begin{pmatrix} a_x & b_x \\ a_y & b_y \end{pmatrix} \begin{pmatrix} x_k \\ y_k \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \end{pmatrix} \end{aligned}$$

This leads to the two overdetermined linear system $\mathbf{A}\mathbf{x}_x = \mathbf{b}_x$ and $\mathbf{A}\mathbf{x}_y = \mathbf{b}_y$ where

$$\mathbf{A} = \begin{pmatrix} x_0 & y_0 & 1 \\ \vdots & \vdots & \vdots \\ x_n & y_n & 1 \end{pmatrix}, \mathbf{x}_x = \begin{pmatrix} a_x \\ b_x \\ t_x \end{pmatrix}, \mathbf{x}_y = \begin{pmatrix} a_y \\ b_y \\ t_y \end{pmatrix}, \mathbf{b}_x = \begin{pmatrix} x'_0 \\ \vdots \\ x'_n \end{pmatrix} \text{ and } \mathbf{b}_y = \begin{pmatrix} y'_0 \\ \vdots \\ y'_n \end{pmatrix}$$

Since \mathbf{A} is not square and therefore not invertible, the solutions of these systems are: $\mathbf{x}_x = \mathbf{A}^\dagger \mathbf{b}_x$ and $\mathbf{x}_y = \mathbf{A}^\dagger \mathbf{b}_y$ where $\mathbf{A}^\dagger = (\mathbf{A}^\top \mathbf{A})^{-1} \mathbf{A}^\top$ is the pseudo-inverse of \mathbf{A} .

Note that in order to compute the normal strains and the shear strain, we only need the four parameters a_x, b_x, a_y and b_y of the affine transformation since we have:

$$\frac{\partial \mathcal{X}'}{\partial x} = a_x, \quad \frac{\partial \mathcal{X}'}{\partial y} = b_x, \quad \frac{\partial \mathcal{Y}'}{\partial x} = a_y \quad \text{and} \quad \frac{\partial \mathcal{Y}'}{\partial y} = b_y$$

leading to the expressions:

$$E_{xx} = \frac{a_x^2 + a_y^2 - 1}{2}, \quad E_{yy} = \frac{b_x^2 + b_y^2 - 1}{2} \quad \text{and} \quad E_{xy} = \frac{a_x b_x + a_y b_y}{2} \quad (4.4)$$

This is because deformation is invariant to rigid-body motions and *a fortiori* to a translation $\begin{pmatrix} t_x & t_y \end{pmatrix}^\top$. The affine transformation can then be represented by only 4 parameters instead of 6.

Local approximation of the displacement functions by a quadratic transformation: Similarly to the approximation by an affine transformation, let's consider a subset of n points \mathbf{m}_k around the point \mathbf{m} (see Fig. 4.3) where we want to compute the strains. The displacement functions over this area can be expressed locally as a quadratic transformation:

$$\forall k = 1 \dots n, \begin{pmatrix} x'_k \\ y'_k \end{pmatrix} = \begin{pmatrix} \mathcal{X}'(x_k, y_k) \\ \mathcal{Y}'(x_k, y_k) \end{pmatrix} \approx \begin{pmatrix} a_x & b_x \\ a_y & b_y \end{pmatrix} \begin{pmatrix} x_k \\ y_k \end{pmatrix} + \begin{pmatrix} c_x \\ c_y \end{pmatrix} x_k y_k + \begin{pmatrix} d_x & e_x \\ d_y & e_y \end{pmatrix} \begin{pmatrix} x_k^2 \\ y_k^2 \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \end{pmatrix} \quad (4.5)$$

The equation system is no more linear here and the parameters $a_x, a_y, b_x, b_y, c_x, c_y, d_x, d_y, e_x, e_y, t_x, t_y$ have now to be estimated using a non-linear optimization procedure. An iterative method such as the Levenberg-Marquardt algorithm is used to minimize the sum of square differences between $\begin{pmatrix} x'_k & y'_k \end{pmatrix}^\top$ and Equ. (4.5).

Compared to the affine approximation, the expression of the derivatives of the displacement functions and particularly the strains are a little more complex when approximated by a quadratic transformation:

$$\begin{aligned} \frac{\partial \mathcal{X}'}{\partial x} &= a_x + c_x y + 2d_x x & \frac{\partial \mathcal{X}'}{\partial y} &= b_x + c_x x + 2e_x y \\ \frac{\partial \mathcal{Y}'}{\partial x} &= a_y + c_y y + 2d_y x & \frac{\partial \mathcal{Y}'}{\partial y} &= b_y + c_y x + 2e_y y \end{aligned}$$

leading to the following expressions for the normal and shear strains:

$$\begin{aligned}
 E_{xx} &= 2(d_x^2 + d_y^2)x^2 + \frac{c_x^2 + c_y^2}{2}y^2 + 2(c_x d_x + c_y d_y)xy + 2(a_x d_x + a_y d_y)x + (a_x c_x + a_y c_y)y \\
 &\quad + \frac{a_x^2 + a_y^2 - 1}{2} \\
 E_{yy} &= \frac{c_x^2 + c_y^2}{2}x^2 + 2(e_x^2 + e_y^2)y^2 + 2(c_x e_x + c_y e_y)xy + (b_x c_x + b_y c_y)x + 2(b_x e_x + b_y e_y)y \\
 &\quad + \frac{b_x^2 + b_y^2 - 1}{2} \\
 E_{xy} &= (c_x d_x + c_y d_y)x^2 + (c_x e_x + c_y e_y)y^2 + \left(2(d_x e_x + d_y e_y) + \frac{c_x^2 + c_y^2}{2}\right)xy \\
 &\quad + \left(b_x d_x + b_y d_y + \frac{a_x c_x + a_y c_y}{2}\right)x + \left(a_x e_x + a_y e_y + \frac{b_x c_x + b_y c_y}{2}\right)y + \frac{a_x b_x + a_y b_y}{2}
 \end{aligned}$$

Note that, similarly to the affine approximation, the translation parameters t_x and t_y are obviously again absent from the expressions of the strains since the deformation is invariant to rigid-body motions. The quadratic transformation can then be represented by 10 parameters instead of 12.

Global approximation of the displacement functions by a spline: displacement functions can be approximated over the whole area of interest using spline functions. The procedure for computing strains is still the same:

1. Fit globally the two functions \mathcal{X}' and \mathcal{Y}' respectively with the splines $\mathcal{S}_{\mathcal{X}'}$ and $\mathcal{S}_{\mathcal{Y}'}$.
2. Compute locally the derivatives $\frac{\partial \mathcal{S}_{\mathcal{X}'}}{\partial x}$, $\frac{\partial \mathcal{S}_{\mathcal{Y}'}}{\partial y}$, $\frac{\partial \mathcal{S}_{\mathcal{Y}'}}{\partial x}$ and $\frac{\partial \mathcal{S}_{\mathcal{X}'}}{\partial y}$ for the given points.
3. Compute E_{xx} , E_{yy} and E_{xy} .

Spline fitting consists in approximating a function with a spline function. The complete process is detailed in Section 2.3.1, Step 3d.

Evaluating the derivative of a spline along the x -axis (respectively y -axis) is equivalent to evaluate the spline with the kernel (basis function) for x -axis (respectively y -axis) replaced by its derivative. Indeed:

$$\mathcal{S}(x, y, \boldsymbol{\alpha}) = \sum_{j=0}^{n_j} \sum_{k=0}^{n_k} \alpha_{j,k} M_j(x) M_k(y) \Rightarrow \begin{cases} \frac{\partial \mathcal{S}(x, y, \boldsymbol{\alpha})}{\partial x} = \sum_{j=0}^{n_j} \sum_{k=0}^{n_k} \alpha_{j,k} \frac{\partial M_j}{\partial x} M_k(y) \\ \frac{\partial \mathcal{S}(x, y, \boldsymbol{\alpha})}{\partial y} = \sum_{j=0}^{n_j} \sum_{k=0}^{n_k} \alpha_{j,k} M_j(x) \frac{\partial M_k}{\partial y} \end{cases}$$

Actually, to simplify the expressions of the kernels, the spacing between knots is considered constant (uniform spline) and equal to 1. x and y are then replaced by $f_x = \frac{x}{k_x}$ and $f_y = \frac{y}{k_y}$ where k_x is the width of the patches and k_y the height. The equations become:

$$\mathcal{S}(x, y, \boldsymbol{\alpha}) = \sum_{j=0}^{n_j} \sum_{k=0}^{n_k} \alpha_{j,k} N_j(f_x) N_k(f_y) \Rightarrow \begin{cases} \frac{\partial \mathcal{S}(x, y, \boldsymbol{\alpha})}{\partial x} = \frac{1}{k_x} \sum_{j=0}^{n_j} \sum_{k=0}^{n_k} \alpha_{j,k} \frac{\partial N_j}{\partial f_x} N_k(f_y) \\ \frac{\partial \mathcal{S}(x, y, \boldsymbol{\alpha})}{\partial y} = \frac{1}{k_y} \sum_{j=0}^{n_j} \sum_{k=0}^{n_k} \alpha_{j,k} N_j(f_x) \frac{\partial N_k}{\partial f_y} \end{cases}$$

Expressions of the basis functions and their derivatives depending on the degree of spline are given in Table 4.1.

Degree of spline	Kernel: $N_*(f)$	Derivative: $\frac{\partial N_*(f)}{\partial f}$
1	$N_i(f) = 1 - f$ $N_{i+1}(f) = f$	$\frac{\partial N_i}{\partial f} = -1$ $\frac{\partial N_{i+1}}{\partial f} = 1$
2	$N_i(f) = \frac{1-2f+f^2}{2} = \frac{(1-f)^2}{2}$ $N_{i+1}(f) = \frac{1+2f-2f^2}{2}$ $N_{i+2}(f) = \frac{f^2}{2}$	$\frac{\partial N_i}{\partial f} = f - 1$ $\frac{\partial N_{i+1}}{\partial f} = 1 - 2f$ $\frac{\partial N_{i+2}}{\partial f} = f$
3	$N_i(f) = \frac{1-3f+3f^2-f^3}{6} = \frac{(1-f)^3}{6}$ $N_{i+1}(f) = \frac{4-6f^2+3f^3}{6}$ $N_{i+2}(f) = \frac{1+3f+3f^2-3f^3}{6}$ $N_{i+3}(f) = \frac{f^3}{6}$	$\frac{\partial N_i}{\partial f} = \frac{-1+2f-f^2}{2} = -\frac{(1-f)^2}{2}$ $\frac{\partial N_{i+1}}{\partial f} = \frac{-4f+3f^2}{2}$ $\frac{\partial N_{i+2}}{\partial f} = \frac{1+2f-3f^2}{2}$ $\frac{\partial N_{i+3}}{\partial f} = \frac{f^2}{2}$
4	$N_i(f) = \frac{1-4f+6f^2-4f^3+f^4}{24} = \frac{(1-f)^4}{24}$ $N_{i+1}(f) = \frac{11-12f-6f^2+12f^3-4f^4}{24}$ $N_{i+2}(f) = \frac{11+12f-6f^2-12f^3+6f^4}{24}$ $N_{i+3}(f) = \frac{1+4f+6f^2+4f^3-4f^4}{24}$ $N_{i+4}(f) = \frac{f^4}{24}$	$\frac{\partial N_i}{\partial f} = \frac{-1+3f-3f^2+f^3}{6} = -\frac{(1-f)^3}{6}$ $\frac{\partial N_{i+1}}{\partial f} = \frac{-3-3f+9f^2-4f^3}{6}$ $\frac{\partial N_{i+2}}{\partial f} = \frac{1-f-3f^2+2f^3}{2}$ $\frac{\partial N_{i+3}}{\partial f} = \frac{1+3f+3f^2-4f^3}{6}$ $\frac{\partial N_{i+4}}{\partial f} = \frac{f^3}{6}$
5	$N_i(f) = \frac{1-5f+10f^2-10f^3+5f^4-f^5}{120} = \frac{(1-f)^5}{120}$ $N_{i+1}(f) = \frac{26-50f+20f^2+20f^3-20f^4+5f^5}{120}$ $N_{i+2}(f) = \frac{66-60f^2+30f^4-10f^5}{120}$ $N_{i+3}(f) = \frac{26+50f+20f^2-20f^3-20f^4+10f^5}{120}$ $N_{i+4}(f) = \frac{1+5f+10f^2+10f^3+5f^4-5f^5}{120}$ $N_{i+5}(f) = \frac{f^5}{120}$	$\frac{\partial N_i}{\partial f} = \frac{-1+4f-6f^2+4f^3-f^4}{24} = -\frac{(1-f)^4}{24}$ $\frac{\partial N_{i+1}}{\partial f} = \frac{-10+8f+12f^2-16f^3+5f^4}{24}$ $\frac{\partial N_{i+2}}{\partial f} = \frac{-12f+12f^3-5f^4}{12}$ $\frac{\partial N_{i+3}}{\partial f} = \frac{5+4f-6f^2-8f^3+5f^4}{12}$ $\frac{\partial N_{i+4}}{\partial f} = \frac{1+4f+6f^2+4f^3-5f^4}{24}$ $\frac{\partial N_{i+5}}{\partial f} = \frac{f^4}{24}$

Table 4.1 — Expressions of the kernels and their respective derivative depending on the degree of spline

4.2 Three-dimensional Strain Measurement

The procedure for three-dimensional strain measurement is similar to the one for two-dimensional strain measurement (see Section 4.1): first, a calibration phase of the imaging sensors (one or more imaging sensors can be used), and secondly, the acquisition of at least two images of the studied object per state of deformation. Then, 3D reconstructions of the object before and after deformation are performed and 3D displacement measurements (leading later to the surface strain) are obtained from the difference of the 3D shapes. But the difference cannot be computed directly because we first need to know for each 3D point of the undeformed shape, which 3D point of the deformed shape is corresponding. In addition to the point correspondences between projected images representing the same state of deformation (we will call them “spatial matches”) and necessary for the 3D reconstruction, it is therefore required to have point correspondences between images of the initial state and images of the deformed state (called “temporal matches”).

Fig. 4.4 presents the classical approach of 3D displacement measurement using DIC [KJC90, LCSP93, SS99, GOP02]. We consider for convenience the simple case of two undeformed and two deformed images:

- ① **Spatial matching between the two images at the initial state:** the corresponding point m'_{ini} of the point m_{ini} is searched into the second image of the initial state using correlation. Then, using the two matches and the projection model parameters obtained by calibration, the 3D point M_{ini} can be obtained by triangulation.
- ② **Temporal matching between the first image of the initial state and the first image of the deformed state:** the corresponding point m_{def} of the point m_{ini} is searched into the first image of the deformed state using correlation.
- ③ **Spatial matching between the two images at the deformed state:** the corresponding point m'_{def} of the point m_{def} is searched into the second image of the deformed state. The deformed 3D point M_{def} can then be obtained by triangulation.
- ④ **Displacement computation:** the 3D displacement is determined as the subtraction of the undeformed 3D point M_{ini} to the second 3D point M_{def} (after deformation).

This procedure is obviously repeated for each point of the undeformed surface, in order to obtain a complete 3D displacement field. This procedure is the commonly used approach even if it presents a known little trouble: the point found by correlation in the second image of the deformed state (Step ③) corresponds to a point already found by correlation (Step ②) and therefore the correlation errors for this point are cumulated. For this reason, Step ③ is sometimes replaced by a direct matching between the first image at the undeformed state I_1 and the second image at the deformed state D_2 .

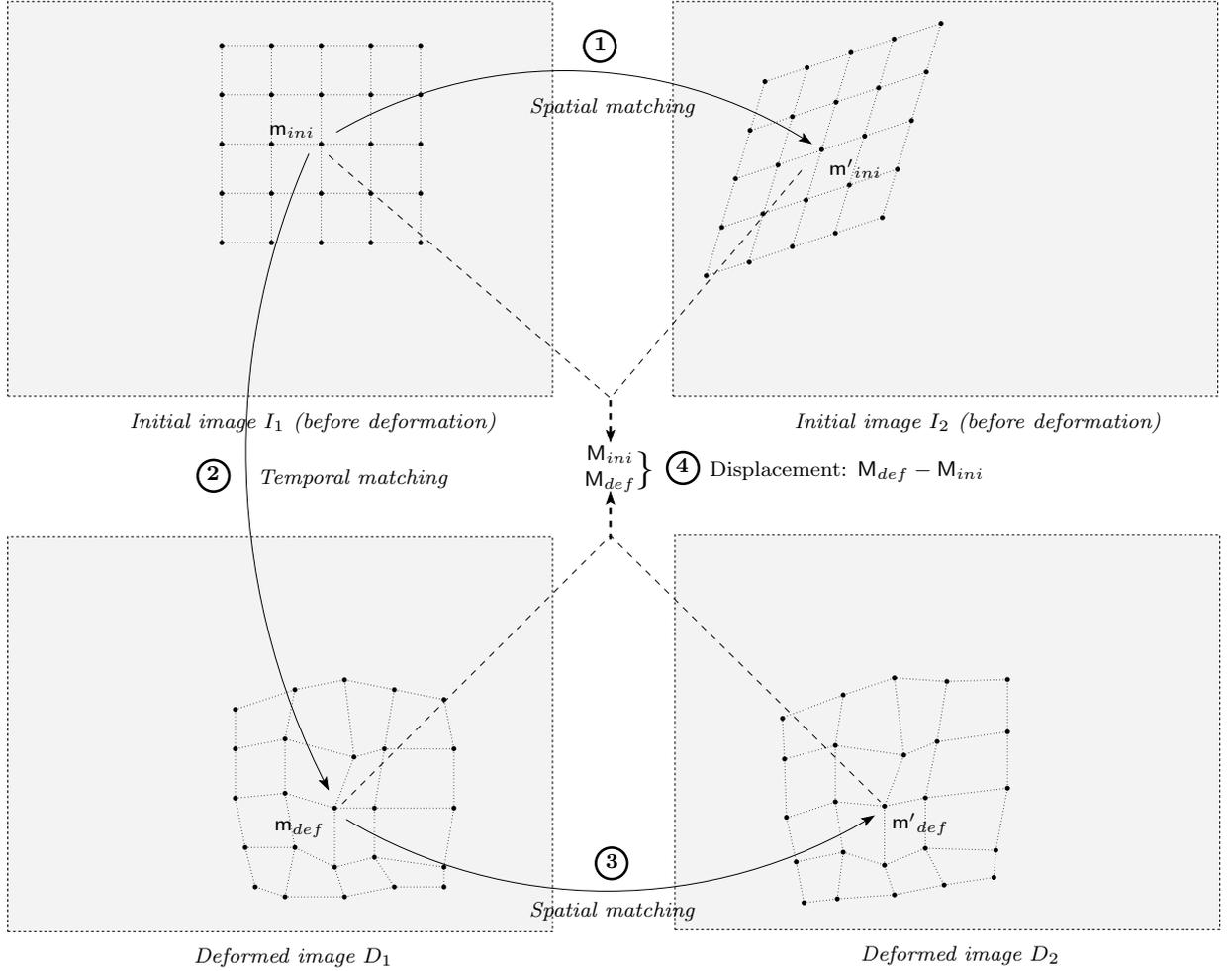


Fig. 4.4 — Classical approach of 3D displacement measurement.

4.2.1 Experimental Issues due to the SEM Imaging System

Application of this approach to our case using a SEM imaging system is possible but new issues are raised. First, and similarly to in-plane strain (see Section 4.1), there is a limitation on the rate of deformation but it is even more restrictive: the deformation process must be slow enough so that at least two images of the specimen can be acquired from different viewpoints. Secondly, and most importantly, the fact that the SEM provides a unique sensor causes a major problem for the computation of the displacement in Step ④. Indeed, as illustrated in Fig. 4.5, the experimental process with a unique sensor consists in moving the specimen in order to acquire at least two images from different viewpoints for each state (initial and deformed state). Because the relative motions undergone by the specimen on the SEM stage between each acquisition are not accurately known, they have to be recovered from image analysis using the videogrammetry approach. However, we have seen in Section 3.4 that only the rotation and the direction of the translation of

the motion undergone can be recovered. Since the magnitude of the translation is only known up to a scale factor, the 3D reconstruction will include as well the unknown scale factor: from the images before deformation, the initial 3D shape will be reconstructed up to λ_{ini} and from the images after deformation, the deformed 3D shape will be known up to a different scale factor λ_{def} . Note that this problem does not occur with systems providing multiple sensors rigidly mounted (such as a stereo-rig of cameras) because the rigid-body transformation between the sensors is always the same during the entire experiment and the unknown scale factor in the 3D reconstruction is therefore constant. In this case, the value of this scale factor can easily be determined using a known distance in the scene at the initial or deformed state but it is not required since strain is a relative measurement (the strain is the same if both 3D shapes are scaled by a same scale factor).

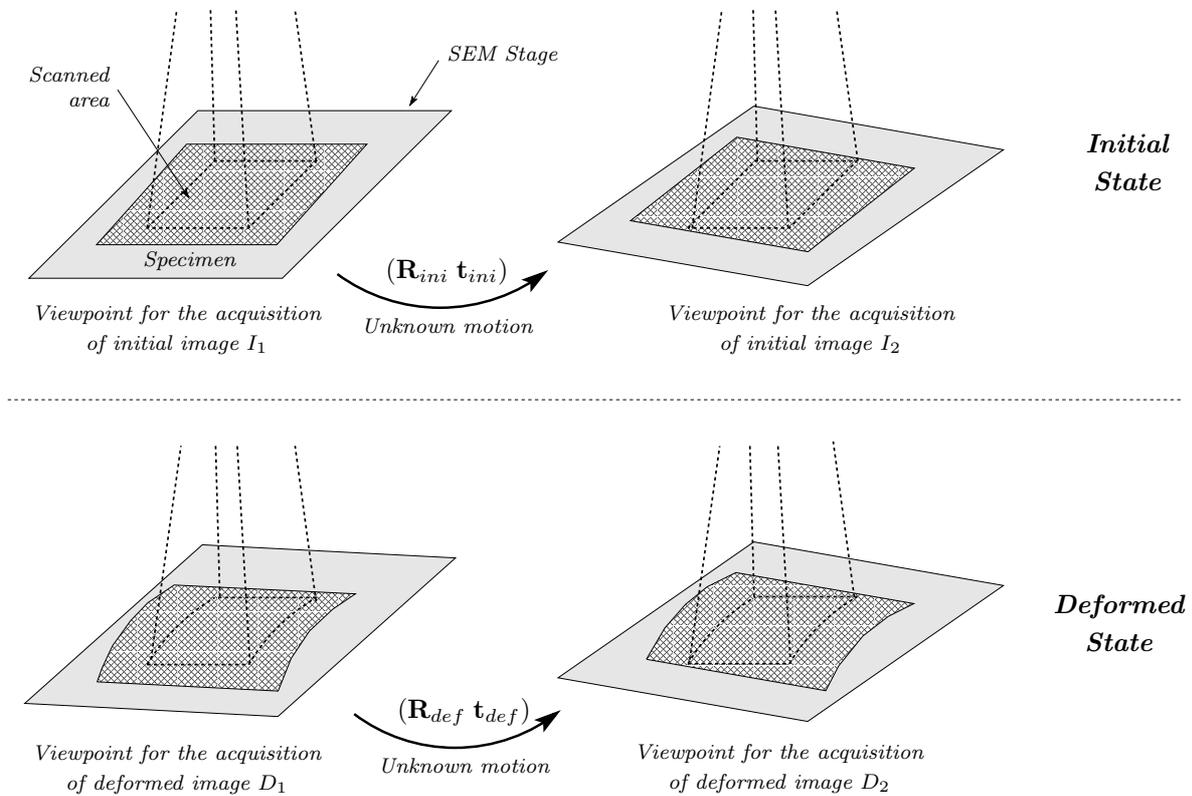


Fig. 4.5 — *Three-dimensional strain measurement with an imaging system providing only one sensor leads to a major problem: because the rigid-body transformation between initial images and the one between deformed images are different, initial and deformed 3D shapes will be reconstructed up to different scale factors and therefore, the 3D displacement field cannot be computed directly.*

To compute the 3D displacement field between initial and deformed state, both 3D shapes obviously need to be expressed with a common scale factor and the ratio $\frac{\lambda_{ini}}{\lambda_{def}}$ is

therefore absolutely necessary. To determine this ratio, different experimental solutions related to the knowledge of specific information of the scene can be investigated:

- comparison of an unknown fixed length in the different 3D reconstructions (using for example the presence of a non-deformable object in the field of view) allows to obtain the needed ratio (equal to the ratio of lengths).
- same statement as above but comparing two unknown fixed areas. It allows to minimize the error on the ratio estimation because several points are used. The needed ratio is equal to the square root of the ratio of areas.
- knowledge of a distance or an area in the initial and deformed 3D reconstruction allows to directly eliminate the two unknown scale factors λ_{ini} and λ_{def} and therefore to be able to compute the 3D displacement.

4.2.2 Strain Tensor Computation

Let's note $\mathbf{M}_i = \begin{pmatrix} X_i & Y_i & Z_i \end{pmatrix}^\top$ the set of undeformed points and $\mathbf{M}'_i = \begin{pmatrix} X'_i & Y'_i & Z'_i \end{pmatrix}^\top$ the set of deformed points (see Fig. 4.6). Analogously to the 2D case (see Section 4.1.1), points in different states are related by the function \mathcal{U} , \mathcal{V} and \mathcal{W} representing respectively the value of the displacement along x , y and z -axis. Respective displacement functions are denoted \mathcal{X}' , \mathcal{Y}' and \mathcal{Z}' :

$$\begin{aligned} \forall i, \quad X'_i &= X_i + \mathcal{U}(X_i, Y_i, Z_i) = \mathcal{X}'(X_i, Y_i, Z_i) \\ Y'_i &= Y_i + \mathcal{V}(X_i, Y_i, Z_i) = \mathcal{Y}'(X_i, Y_i, Z_i) \\ Z'_i &= Z_i + \mathcal{W}(X_i, Y_i, Z_i) = \mathcal{Z}'(X_i, Y_i, Z_i) \end{aligned}$$

The different tensors used in strain computation have already been detailed in Section 4.1.1 and only the mathematical expression will be given in this section. The deformation gradient tensor \mathbf{F} in 3D is expressed as follows:

$$\begin{aligned} \mathbf{F} &= \begin{pmatrix} \frac{\partial \mathcal{X}'}{\partial X} & \frac{\partial \mathcal{X}'}{\partial Y} & \frac{\partial \mathcal{X}'}{\partial Z} \\ \frac{\partial \mathcal{Y}'}{\partial X} & \frac{\partial \mathcal{Y}'}{\partial Y} & \frac{\partial \mathcal{Y}'}{\partial Z} \\ \frac{\partial \mathcal{Z}'}{\partial X} & \frac{\partial \mathcal{Z}'}{\partial Y} & \frac{\partial \mathcal{Z}'}{\partial Z} \end{pmatrix} \\ &= \begin{pmatrix} 1 + \frac{\partial \mathcal{U}'}{\partial X} & \frac{\partial \mathcal{U}'}{\partial Y} & \frac{\partial \mathcal{U}'}{\partial Z} \\ \frac{\partial \mathcal{V}'}{\partial X} & 1 + \frac{\partial \mathcal{V}'}{\partial Y} & \frac{\partial \mathcal{V}'}{\partial Z} \\ \frac{\partial \mathcal{W}'}{\partial X} & \frac{\partial \mathcal{W}'}{\partial Y} & 1 + \frac{\partial \mathcal{W}'}{\partial Z} \end{pmatrix} \end{aligned}$$

leading to the following Green-Lagrange strain tensor \mathbf{E} :

$$\mathbf{E} = \begin{pmatrix} E_{xx} & E_{xy} & E_{xz} \\ E_{xy} & E_{yy} & E_{yz} \\ E_{xz} & E_{yz} & E_{zz} \end{pmatrix}$$

where:

$$E_{xx} = \frac{1}{2} \left[\left(\frac{\partial \mathcal{X}'}{\partial X} \right)^2 + \left(\frac{\partial \mathcal{Y}'}{\partial X} \right)^2 + \left(\frac{\partial \mathcal{Z}'}{\partial X} \right)^2 - 1 \right] = \frac{\partial \mathcal{U}}{\partial X} + \frac{1}{2} \left[\left(\frac{\partial \mathcal{U}}{\partial X} \right)^2 + \left(\frac{\partial \mathcal{V}}{\partial X} \right)^2 + \left(\frac{\partial \mathcal{W}}{\partial X} \right)^2 \right]$$

$$E_{yy} = \frac{1}{2} \left[\left(\frac{\partial \mathcal{X}'}{\partial Y} \right)^2 + \left(\frac{\partial \mathcal{Y}'}{\partial Y} \right)^2 + \left(\frac{\partial \mathcal{Z}'}{\partial Y} \right)^2 - 1 \right] = \frac{\partial \mathcal{V}}{\partial Y} + \frac{1}{2} \left[\left(\frac{\partial \mathcal{U}}{\partial Y} \right)^2 + \left(\frac{\partial \mathcal{V}}{\partial Y} \right)^2 + \left(\frac{\partial \mathcal{W}}{\partial Y} \right)^2 \right]$$

$$E_{zz} = \frac{1}{2} \left[\left(\frac{\partial \mathcal{X}'}{\partial Z} \right)^2 + \left(\frac{\partial \mathcal{Y}'}{\partial Z} \right)^2 + \left(\frac{\partial \mathcal{Z}'}{\partial Z} \right)^2 - 1 \right] = \frac{\partial \mathcal{W}}{\partial Z} + \frac{1}{2} \left[\left(\frac{\partial \mathcal{U}}{\partial Z} \right)^2 + \left(\frac{\partial \mathcal{V}}{\partial Z} \right)^2 + \left(\frac{\partial \mathcal{W}}{\partial Z} \right)^2 \right]$$

$$E_{xy} = \frac{1}{2} \left(\frac{\partial \mathcal{X}'}{\partial X} \frac{\partial \mathcal{X}'}{\partial Y} + \frac{\partial \mathcal{Y}'}{\partial X} \frac{\partial \mathcal{Y}'}{\partial Y} + \frac{\partial \mathcal{Z}'}{\partial X} \frac{\partial \mathcal{Z}'}{\partial Y} \right) = \frac{1}{2} \left(\frac{\partial \mathcal{U}}{\partial Y} + \frac{\partial \mathcal{V}}{\partial X} + \frac{\partial \mathcal{U}}{\partial X} \frac{\partial \mathcal{U}}{\partial Y} + \frac{\partial \mathcal{V}}{\partial X} \frac{\partial \mathcal{V}}{\partial Y} + \frac{\partial \mathcal{W}}{\partial X} \frac{\partial \mathcal{W}}{\partial Y} \right)$$

$$E_{xz} = \frac{1}{2} \left(\frac{\partial \mathcal{X}'}{\partial X} \frac{\partial \mathcal{X}'}{\partial Z} + \frac{\partial \mathcal{Y}'}{\partial X} \frac{\partial \mathcal{Y}'}{\partial Z} + \frac{\partial \mathcal{Z}'}{\partial X} \frac{\partial \mathcal{Z}'}{\partial Z} \right) = \frac{1}{2} \left(\frac{\partial \mathcal{U}}{\partial Z} + \frac{\partial \mathcal{W}}{\partial X} + \frac{\partial \mathcal{U}}{\partial X} \frac{\partial \mathcal{U}}{\partial Z} + \frac{\partial \mathcal{V}}{\partial X} \frac{\partial \mathcal{V}}{\partial Z} + \frac{\partial \mathcal{W}}{\partial X} \frac{\partial \mathcal{W}}{\partial Z} \right)$$

$$E_{yz} = \frac{1}{2} \left(\frac{\partial \mathcal{X}'}{\partial Y} \frac{\partial \mathcal{X}'}{\partial Z} + \frac{\partial \mathcal{Y}'}{\partial Y} \frac{\partial \mathcal{Y}'}{\partial Z} + \frac{\partial \mathcal{Z}'}{\partial Y} \frac{\partial \mathcal{Z}'}{\partial Z} \right) = \frac{1}{2} \left(\frac{\partial \mathcal{V}}{\partial Z} + \frac{\partial \mathcal{W}}{\partial Y} + \frac{\partial \mathcal{U}}{\partial Y} \frac{\partial \mathcal{U}}{\partial Z} + \frac{\partial \mathcal{V}}{\partial Y} \frac{\partial \mathcal{V}}{\partial Z} + \frac{\partial \mathcal{W}}{\partial Y} \frac{\partial \mathcal{W}}{\partial Z} \right)$$

Actually, because we do not have access to information under the surface, we only have surface deformation and gradients perpendicular to the surface $\frac{\partial^*}{\partial Z}$ are unknown. Therefore, the values of E_{xz} , E_{yz} and E_{zz} cannot be determined.

Due to lack of time, the three-dimensional strain computation has not been entirely implemented in the thesis. The method is implemented until the 3D displacement field computation (from Step ① to Step ④ of Section 4.2) but the final step (computation of the strain tensor) is not available. For the experimental results of Section 4.3, the strains are computed using Abaqus[®], a software for finite element analysis. The computation is achieved from the 3D displacements given at the nodes of a triangular mesh, generated using *Visiocre*, the computer vision software developed in this work (see Section 5).

However, in order to present the complete method of three-dimensional strain measurement and to give an idea of how the strain tensor can be computed, a simple method commonly used is described in the following sections.

4.2.3 Change of Coordinate System

In the case of three-dimensional strain measurement, the strain in a given measure point is computed by averaging the strains in the triangles formed by the neighboring points. As illustrated in Fig. 4.6, a triangular mesh is then created for each subsets of points (initial and deformed) and deformation is defined as the transformation converting a triangle in the initial state to the corresponding triangle in the deformed state (Lagrangian formulation).

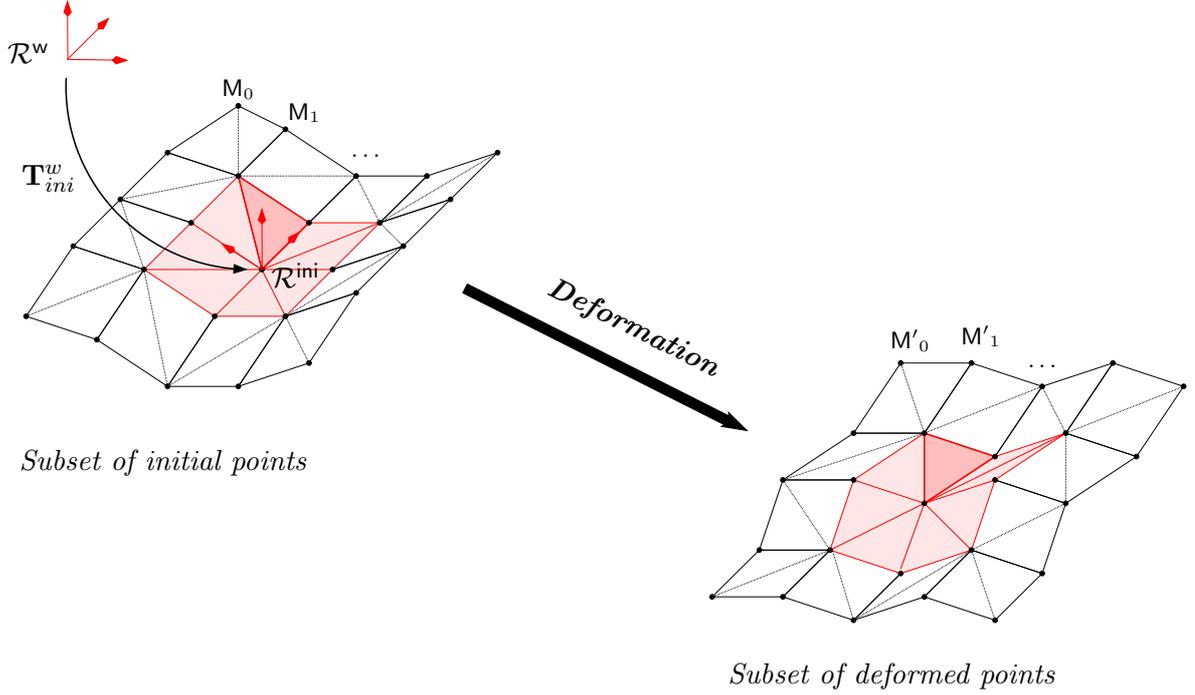


Fig. 4.6 — The set of 3D points M_i are deformed into the set of 3D points M'_i . The three-dimensional strain measurement in a given measure point is achieved by averaging the strain in all the triangles it belongs to. In order to have surface deformation, the coordinate system is changed for each triangle so that this triangle in the initial state lies onto the plane $Z = 0$ and the origin is the measure point.

Because the strain in a given triangle is considered at the surface of the object, it is required to be located on the tangent plane to the surface at the measure point and a change of coordinate system is therefore necessary before approximating the displacement functions and computing the strain associated to the triangle. Actually, in practice, the world coordinate system \mathcal{R}^w is not transformed into the real tangent plane to the surface at the measure point but into the local coordinate system \mathcal{R}^{ini} where the origin is the measure point and the corresponding triangle in the initial state lies onto the plane $Z = 0$. The rigid-body transformation between \mathcal{R}^w and \mathcal{R}^{ini} is denoted $\mathbf{T}_{\text{ini}}^w$ (see Fig. 4.6).

4.2.4 Displacement Functions Approximation

Similarly to two-dimensional strain measurement, the approximation of the displacement functions is necessary because the real functions \mathcal{X}' , \mathcal{Y}' and \mathcal{Z}' are unknown. However, because the strain is computed for each triangle around the measure point, only 3 couples of points are available to estimate the displacement functions and they can only be

approximated by an affine transformation (supposing an homogeneous deformation in this triangle). The approximation procedure for 3D displacement functions is analogous to the developments in 4.1.2 and the expression for the strain tensor components in the case of a three-dimensional displacement can be readily deduced from these equations.

For a given triangle composed of the 3 points M_0 , M_1 and M_2 , the approximation of the displacement functions by an affine transformation consists in determining the 12 parameters $a_x, a_y, a_z, b_x, b_y, b_z, c_x, c_y, c_z, t_x, t_y$ and t_z such that:

$$\forall k = 0 \dots 2, \begin{pmatrix} X'_k \\ Y'_k \\ Z'_k \end{pmatrix} \approx \begin{pmatrix} a_x & b_x & c_x \\ a_y & b_y & c_y \\ a_z & b_z & c_z \end{pmatrix} \begin{pmatrix} \hat{X}_k \\ \hat{Y}_k \\ \hat{Z}_k \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix}$$

where $\hat{M}_k = (\hat{X}_k \ \hat{Y}_k \ \hat{Z}_k)^\top$ are the coordinates of the k^{th} undeformed point in the new local coordinate system: $\hat{M}_k = \mathbf{T}_{ini}^w M_k$ (Section 4.2.3).

Actually, all the initial points \hat{M}_k are in the plane defined by the initial triangle and then $\forall k = 0 \dots 2, \hat{Z}_k = 0$. Therefore, the three parameters c_x, c_y and c_z cannot be estimated (with $\hat{Z}_k = 0$, they disappear from the equations). This is coherent with the fact that the displacements perpendicular to the surface $\frac{\partial^*}{\partial Z}$ are unknown and the values of E_{xz}, E_{yz} and E_{zz} cannot be determined. Finally, the approximation is:

$$\forall k = 0 \dots 2, \begin{pmatrix} X'_k \\ Y'_k \\ Z'_k \end{pmatrix} \approx \begin{pmatrix} a_x & b_x \\ a_y & b_y \\ a_z & b_z \end{pmatrix} \begin{pmatrix} \hat{X}_k \\ \hat{Y}_k \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix}$$

which leads to the following expressions for the normal and shear strains:

$$E_{xx} = \frac{a_x^2 + a_y^2 + a_z^2 - 1}{2} \quad E_{yy} = \frac{b_x^2 + b_y^2 + b_z^2 - 1}{2} \quad E_{xy} = \frac{a_x b_x + a_y b_y + a_z b_z}{2} \quad (4.6)$$

Note that similarly to the two-dimensional strain measurement, the translation parameters t_x, t_y and t_z does not appear in the strain computation since deformation is invariant to rigid-body motions.

4.3 Experimental Results

4.3.1 Experimental Context: Euro vs. Dollar

In order to validate the entire process of calibration, 3D reconstruction and strain computation, a “zero-deformation” experiment is conducted: several images of a specimen undergoing only rigid-body motions are acquired and strains (supposed to be null) are computed between the two 3D shapes reconstructed by videogrammetry. It would have been better to conduct an experiment with a specimen undergoing tensile loading in a SEM loading stage and compare with strain gauges but we do not have any SEM loading stage available at the current time.

After the 3D reconstruction of the letter “R” on a penny (see Section 3.5.1), the experiment is conducted with the letter “R” of the word “EURO” of a coin of 10 cents of Euro (see Fig. 4.7) in order to avoid any jealousy between the two currencies.

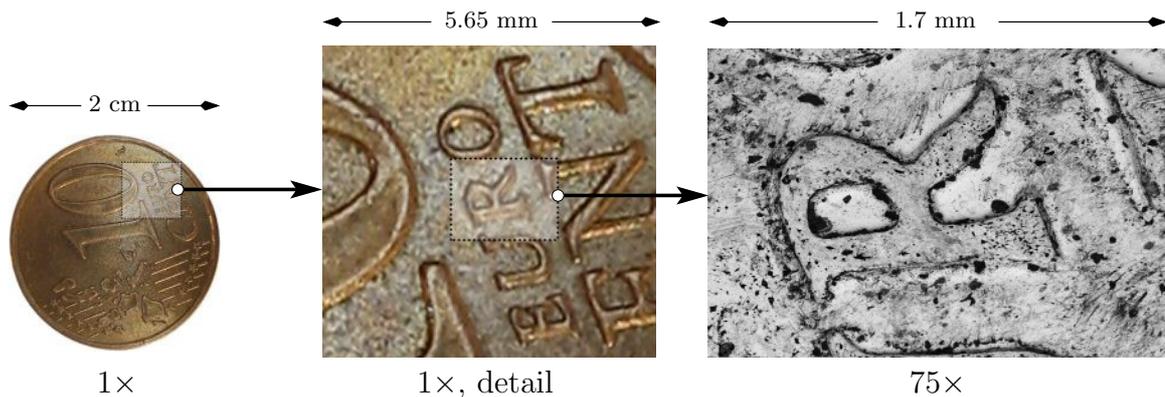


Fig. 4.7 — *The specimen used for the experiment is a coin of 10 cents of Euro. The letter “R” of the word “EURO” is observed with a SEM in order to be reconstructed in 3D.*

The FEI XL 30 ESEM of Ecole des Mines d’Albi-Carmaux is operated in high-vacuum mode (SEM mode). As described in see Section 3.5.1, it is better to work at a low voltage and with the BSE detector. The operating conditions for this experiment are as follows:

- BSE detector
- 75× magnification
- non-dimensional spot-size of 4
- 10 mm working distance
- accelerating voltage of 5 kV

- seven-bit gray scale images (712×484 size)

After the image acquisition, the experiment consists of the four following steps: a) calibration, b) reconstruction of the two 3D shapes and c) 3D displacement measurement and d) strain computation.

4.3.2 Calibration

Due to its texture and its relative flatness, the coin is usable as a quasi-planar calibration target. However, a speckle pattern calibration target realized by micro-lithography is preferred to insure optimal results because its geometry is closer to a plane and its geometry is then better initialized ($Z = 0$). A sequence of 7 images of this target is acquired from different viewpoints (see Fig. 4.8).

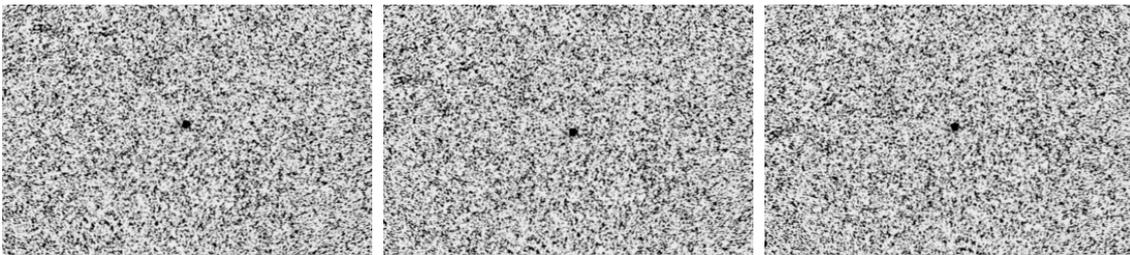


Fig. 4.8 — *Three images from the sequence used for the calibration procedure (speckle pattern textured specimen realized by micro-lithography).*

The experiment is conducted at very low magnification ($75\times$) and similarly to the first experiment with the SEM at low magnification (see Section 2.4.3), the drift is negligible and the time-varying distortion is not corrected. The spatially-varying distortion is fitted by a bicubic spline composed of 10×10 patches and the correction field is illustrated in Fig. 4.9. The distribution of the reprojection errors follows a normal distribution proving that the remaining errors are essentially due to Gaussian noise. Moreover, the calibration results are very good since the standard deviation of the magnitudes of reprojection errors is 0.01 pixels and the bias is 0.02 pixels.

4.3.3 3D Reconstruction

After the calibration, the operating conditions must obviously remain constant until the end of the experiment. In order to avoid to remove the calibration target from the SEM chamber after the image acquisition, the coin of 10 cents of Euro is fixed on the same specimen holder (see Fig. 4.10). By this way, the acquisition of the images of the letter

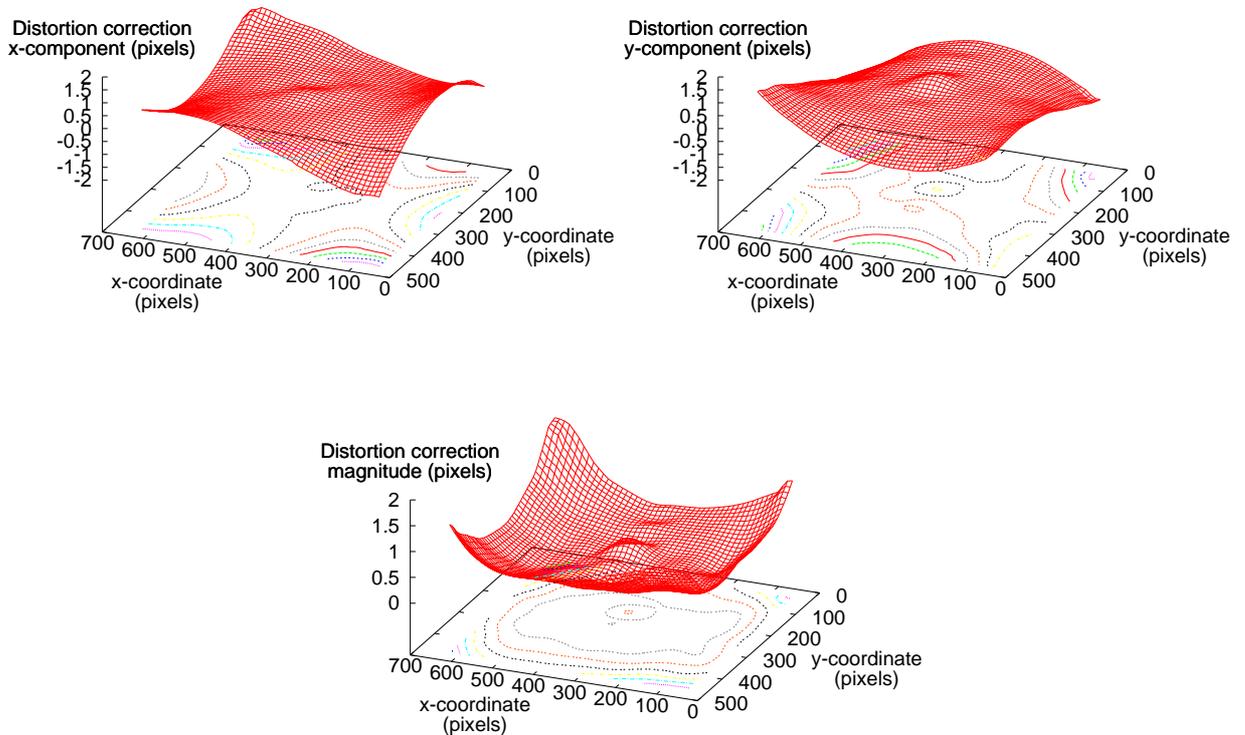


Fig. 4.9 — *Distortion correction field computed using the especially designed speckle pattern calibration target: x-component (left), y-component (right) and magnitude (bottom).*

“R” of the word “EURO” can be done consecutively without changing any parameters of the SEM.

For the 3D shape measurement using videogrammetry, two sets of three images of the coin detail are acquired from different viewpoints (see Fig. 4.11). The three first images are used for the reconstruction of a first 3D shape (considered as the initial state) represented in Fig. 4.12 and the three last images are used for a second 3D reconstruction (considered as the deformed state).

4.3.4 3D Displacement Measurement

In addition to the spatial matchings, a temporal matching is achieved between the first images of each set in order to have a relationship between the 3D points in the initial and deformed states. However, the 3D displacement cannot be estimated directly because the two 3D shapes are reconstructed up to a different scale factor, denoted λ_{ini} for the initial 3D shape and λ_{def} for the deformed one (see Section 4.2). The ratio $\frac{\lambda_{ini}}{\lambda_{def}}$ between the

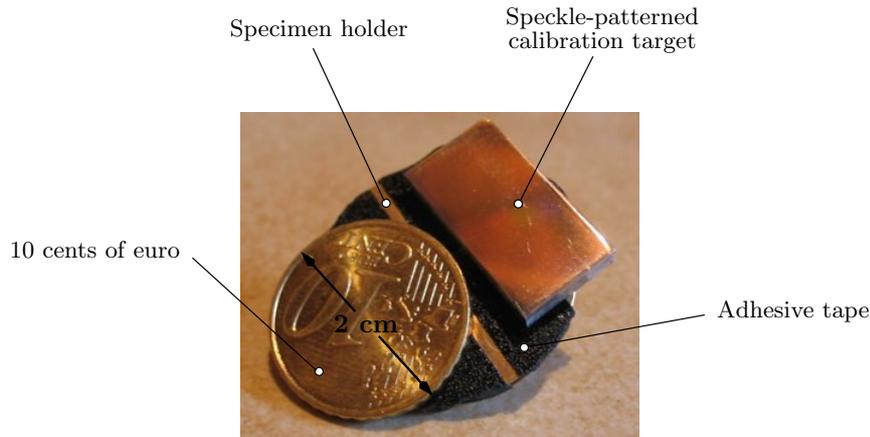


Fig. 4.10 — *Setup of the experiment: the coin is stuck close to the calibration target on the same specimen holder in order to keep the same operating conditions for the acquisition of all images.*

two shapes can be easily determined because the specimen has not actually been deformed since only rigid-body motions have been imposed (zero-deformation). Different methods have been presented in Section 4.2.1 to obtain the ratio. Here, several pairs of 3D point correspondences are used to compare 3D distances which are supposed to be equal. The estimated ratio of scale factors is obviously slightly different depending on the pair of 3D points used and the mean value of the different ratios is finally kept for $\frac{\lambda_{ini}}{\lambda_{def}}$. At last, the second 3D shape is scaled by $\frac{\lambda_{ini}}{\lambda_{def}}$ so that the two shapes are reconstructed up to the same unknown scale factor λ_{ini} . The 3D displacement field is obtained by difference of corresponding 3D points. Note that the actual value of the common scale factor λ_{ini} does not need to be determined because we are computing strain, which is a relative measurement (the strain is the same if both 3D shapes are scaled by a same scale factor).

4.3.5 Strain Computation

The computation of the strain tensor has not been implemented in our programs and the strains in this experiment are estimated using Abaqus[®], a software for finite element analysis. Abaqus[®] needs the 3D displacements at the nodes of a mesh as input but no 3D mesher is implemented in our programs. Then, the module of *Visiocre* of 2D mesh using Delaunay triangulation (see Section 5) is used instead: a 2D triangular mesh is generated from the initial 2D points in the first image and the neighboring relationships between the points are considered equivalent in 3D (see Fig. 4.13). This is obviously not true for complex shapes but it is an acceptable assumption for the shape of the letter “R” in this experiment.

Strain values are obtained with Abaqus[®] considering a thin shell structure and the

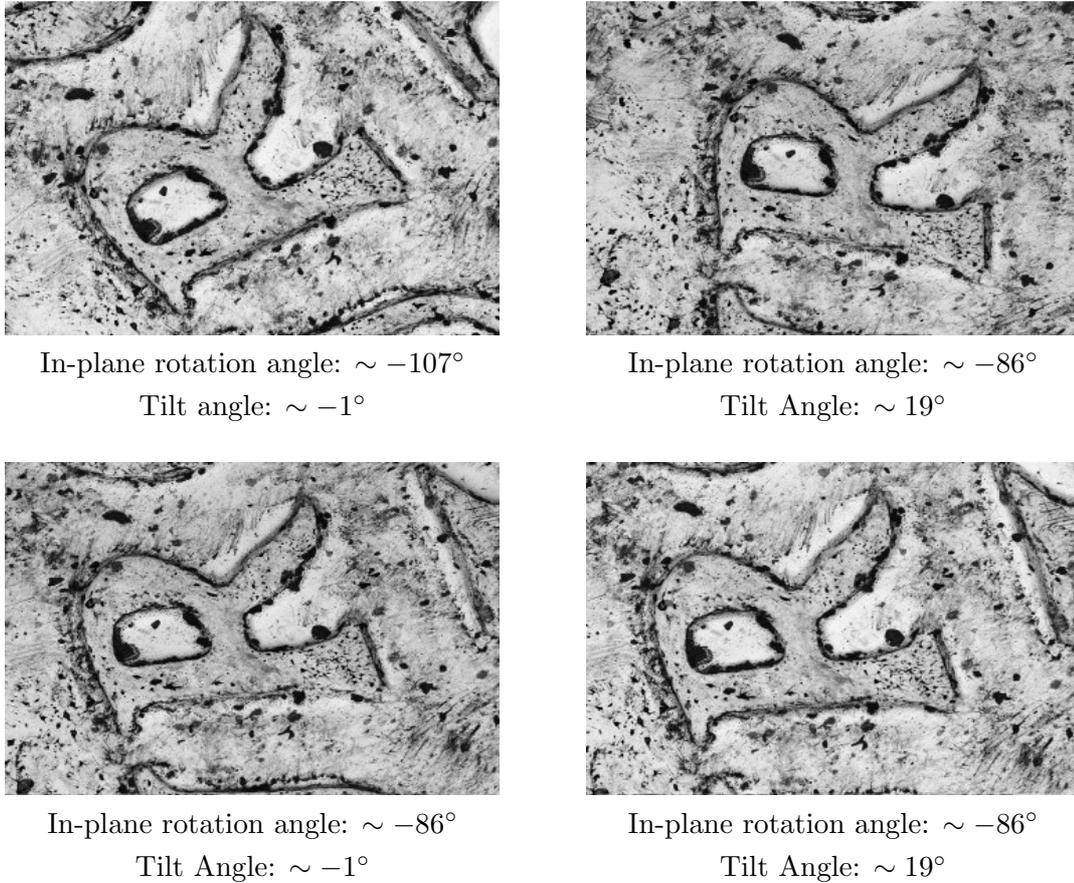


Fig. 4.11 — *Four BSE images of the letter “R” of the 10 cents coin. A total of 6 images are acquired for the two 3D reconstructions.*

nodes of the mesh being defined every 3 pixels. Results are illustrated in Fig. 4.14 and 4.15. Because only rigid-body motions have been undergone between the different image acquisitions, the computed strains should be null. Actually, they are not, and even if the standard deviation and the bias of errors are acceptable (bias = 1%, standard deviation = 1.8% for the maximum principal strain and bias = -0.5%, standard deviation = 1% for the minimum principal strain), the results are a little disappointing because errors can reach 5% for the maximum principal strain and -1.5% for the minimum principal strain.

The errors presented here are due to the different errors cumulated in the 4 steps of the experiment:

- a) errors due to the calibration (see Section 4.3.2)
- b) errors due to the 3D reconstruction (see Section 4.3.3)
- c) errors due to the identification of the ratio of scale factors (see Section 4.3.4)
- d) error due to the method of strain computation (see Section 4.3.5).

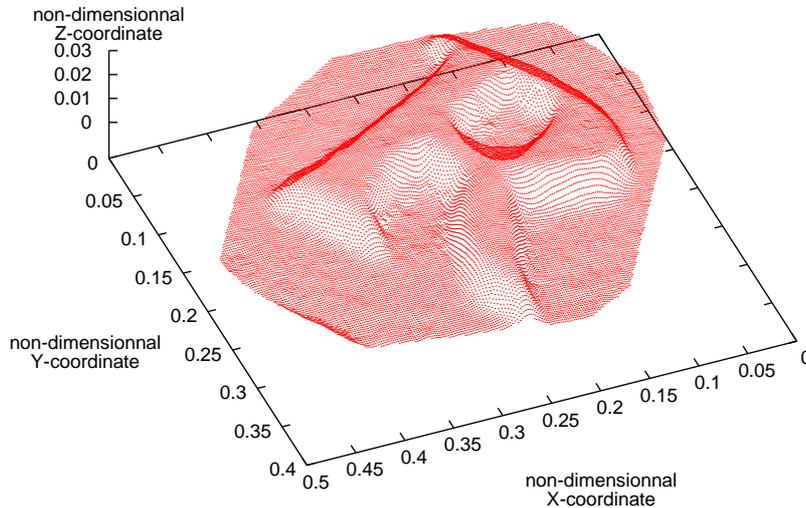


Fig. 4.12 — *One of the two reconstructed 3D shapes of the letter “R” of “EURO”, known up to a scale factor.*

Using our new calibration approach, a) errors due to the calibration are now very small and they certainly not represent the main error component here. Likewise, strain computation being realized using Abaqus[®], d) errors due to the method of computation are not the first cause of error even if the important variations in the strain field (see Figs 4.14, 4.14 and 4.16) are the consequence of a too small element size compared to the data spacing (spacing between each node of the mesh is 3 pixels). A larger spacing of the nodes (15-20 pixels) would lead to a much smoother strain field and then, much better results. However, the main cause of error here is clearly b) the 3D reconstruction. Indeed, we can notice in Fig. 4.14 that the highest strain values are located on the slopes of the letter “R” where the 3D reconstruction is more difficult. This is due to two main reasons: first, the DIC technique is less efficient in areas of high curvature and secondly, the curved areas are subject to occlusion problems when the specimen undergoes rigid-body motions: points of these areas are not present in all the images, leading to a less accurate reconstruction. On the contrary, the results of the strain computation on the areas of low curvature of the specimen are quite good since the errors vary from 0 to 1.5% (see Fig. 4.16). The second main cause of error is c) the identification of the ratio of scale factors. The differences between the different ratios estimated from pairs of 3D points (see Section 4.3.4) is significant (a few percents) and even if the error is minimized by taking the mean value, it does not seem sufficient here. A first improvement could be to use only

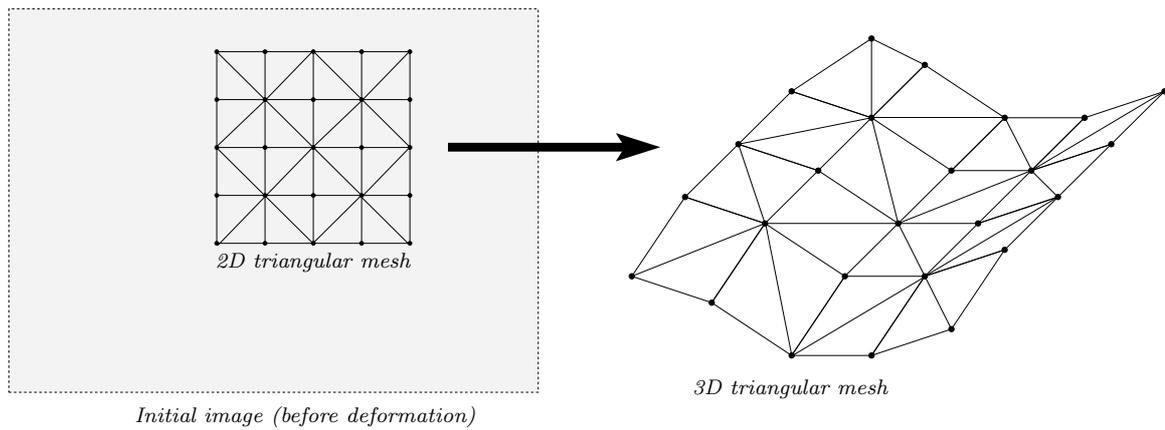


Fig. 4.13 — *The 3D mesh required for Abaqus[®] is obtained using the module of 2D mesh and considering that the neighboring relationships are the same for the 2D points and the 3D points.*

the pairs of the most accurately reconstructed 3D points (corresponding to the smallest reprojection errors).

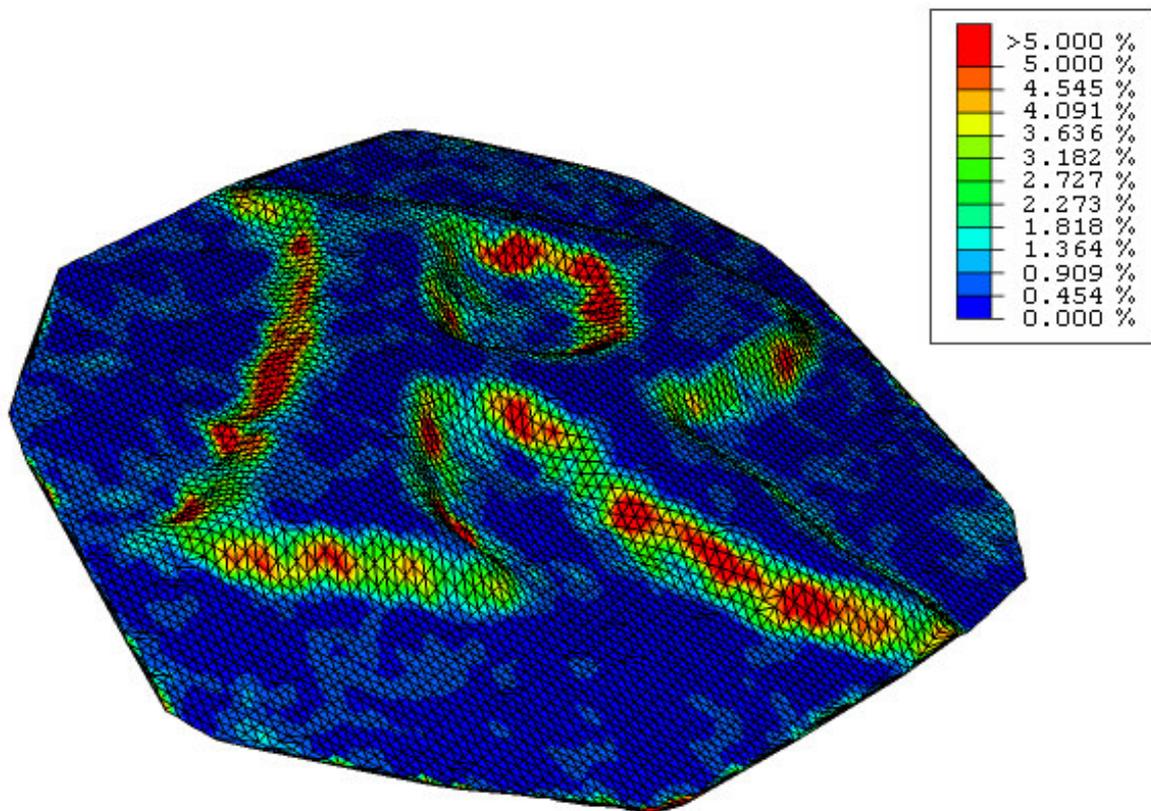


Fig. 4.14 — *Maximum principal strain values obtained for the zero-deformation experiment. The strain values go from 0% to 5%, the highest errors being located on the slopes of the letter “R”.*

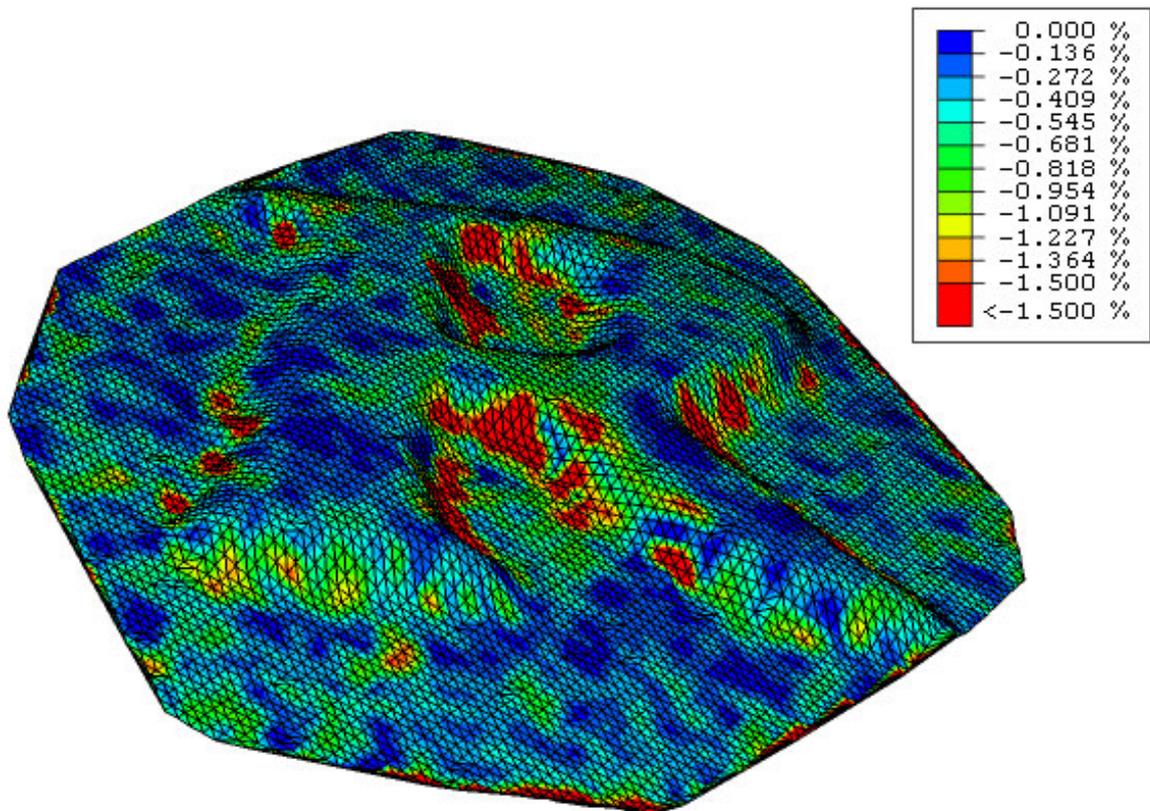


Fig. 4.15 — *Minimum principal strain values obtained for the zero-deformation experiment. The strain values go from -1.5% to 0%.*

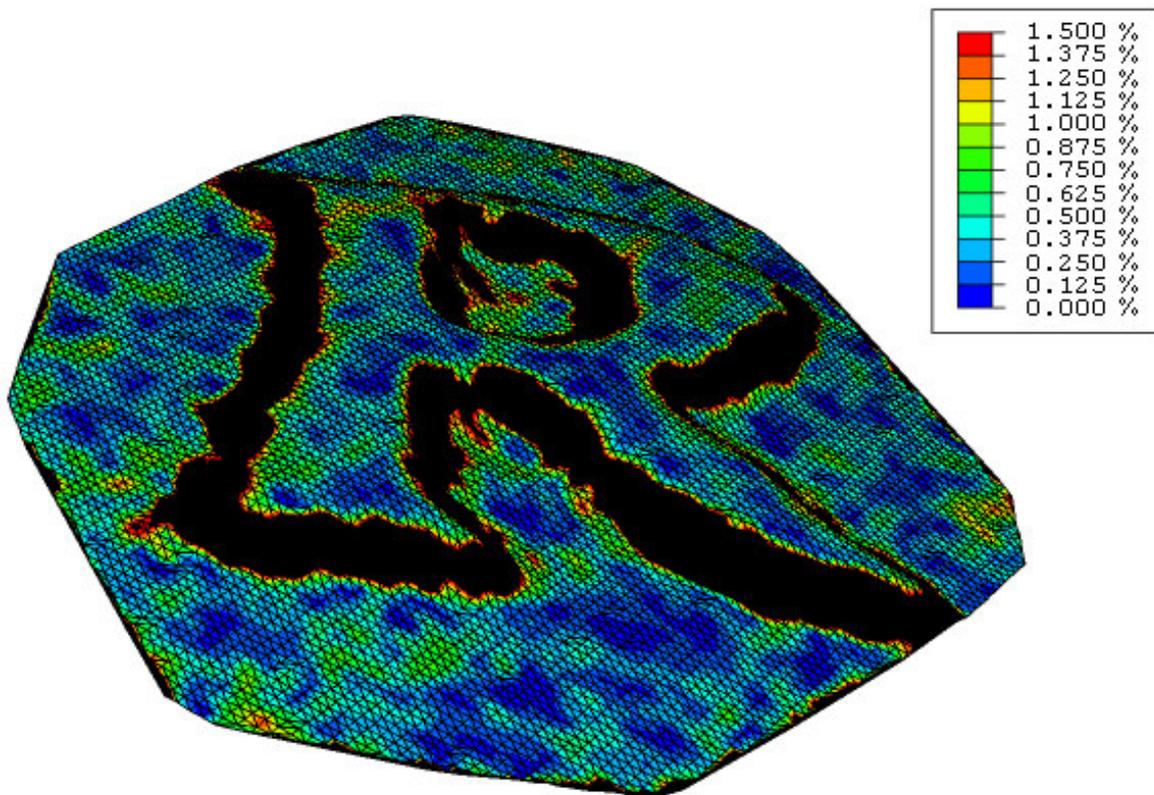


Fig. 4.16 — *Maximum principal strain values considering only the quasi-planar areas of the object. The results are much better here and the errors vary from 0% to 1.5%.*

Bibliography

- [ABBC94] Lucien Allais, Michel Bornert, Thierry Bretheau, and Daniel Caldemaison. Experimental characterization of the local strain field in a heterogeneous elastoplastic material. *Acta Metallurgica et Materialia*, 42(11):3865–3880, 1994.
- [BDM04] M. Bornert, P. Doumalin, and E. Maire. Mesure tridimensionnelle des champs de déformation par corrélation d’images microtomographiques. In *Photomécanique 2004*, pages 33–40, Albi (France), 2004.
- [BSFS99] B.K. Bay, T.S. Smith, D.P. Fyhrie, and M. Saad. Digital volume correlation: three-dimensional strain mapping using X-ray tomography. *Experimental Mechanics*, 39(3):218–227, 1999.
- [CRSP85] T. C. Chu, W. F. Ranson, M. A. Sutton, and W. H. Peters. Application of Digital Image Correlation Techniques to Experimental Mechanics. *Experimental Mechanics*, 25(3):232–244, 1985.
- [Dou00] Pascal Doumalin. *Microextensométrie locale par corrélation d’images numériques*. PhD thesis, Ecole Polytechnique, 2000.
- [GOP02] Dorian Garcia, Jean-José Orteu, and Luc Penazzi. A combined temporal tracking and stereo-correlation technique for accurate measurement of 3D displacements: application to sheet metal forming. *Journal of Materials Processing Technology*, 125-126:736–742, Jan 2002.
- [KJC90] Z. L. Khan-Jetter and T. C. Chu. Three-Dimensional Displacement Measurements Using Digital Image Correlation and Photogrammic Analysis. *Experimental Mechanics*, 30(1):10–16, 1990.
- [LCSP93] P. F. Luo, Y. J. Chao, M. A. Sutton, and W. H. Peters. Accurate Measurement of Three-Dimensional Deformable and Rigid Bodies Using Computer Vision. *Experimental Mechanics*, 33(2):123–132, 1993.
- [MNSW⁺83] S. R. Mc Neill, M. A. Sutton, W. J. Wolters, W. H. Peters, and W. F. Ranson. Determination of Displacements Using an Improved Digital Correlation Method. *Image and Vision Computing*, 1(3):1333–1339, 1983.
- [SCP⁺86] M.A. Sutton, M. Cheng, W. H. Peters, Y. J. Chao, and S. R. McNeill. Application of an Optimized Digital Correlation Method to Planar Deformation Analysis. *Image and Vision Computing*, 4(3):143–153, 1986.

- [SS99] P. Synnergren and M. Sjö Dahl. A Stereoscopic Digital Speckle Photography System for 3D Displacement Field Measurements. *Optics and Lasers in Engineering*, 31:425–443, 1999.
- [SWP⁺83] M. A. Sutton, W. J. Wolters, W. H. Peters, W. F. Ranson, and S. R McNeill. Determination of Displacements Using an Improved Digital Correlation Method. *Image and Vision Computing*, 21:133–139, 1983.

Chapter 5

Software Development

Applications of the videogrammetry are very numerous and in many fields (see [CLVWdV02] for an application to volcanology for instance). Ecole des Mines d'Albi-Carmaux started developing this technique in 1999 for architectural applications [AGO00] and decided in 2001 to apply it for micro- and nano-scale measurements using a SEM [Cor01]. During this thesis, a general modular software gathering different computer vision applications and particularly a module of videogrammetry has been developed. This software being aimed at any user, not especially experienced in computer vision, a particular attention has been paid to the user-friendliness (usability) and consequently to the Graphic User Interface (GUI). The software has been developed under Linux (RedHat distribution) in C++ and is designed for running under different platforms (particularly Windows). In this sense, it uses the portable *Qt* library¹ for the GUI. In addition to the standard C++ library *stdlib*, it uses as well a scientific library *v3d* that we developed for computer vision² and algebra algorithms and a graphic library *GUILib* that we developed to add functionalities of image visualization: zoom, primitive and text drawing, palette change, restriction of mouse movement, crosshair, *etc.* However, the development of a complete software is very time-consuming and does not accommodate with the work required for a thesis. The development of the last applications (mainly spatially-varying and time-varying distortion correction) are then only command-line programs and will not be detailed in this chapter. They have been implemented simultaneously with a new scientific library called *csf*, developed in collaboration with Dorian Garcia and Hubert Schreier at USC (University of South Carolina).

¹*Qt* library is developed by Trolltech: <http://www.trolltech.com>

²The library development of *v3d* was initiated by Dorian Garcia during his PhD thesis [Gar01].

5.1 *Visiocre* Software

The software is called *Visiocre* and has been developed following a generic and modular approach³ so that new computer vision applications can be included easily to the main application. Every module represents a specific application and outputs its results as formatted data to a global repository. This repository gathers all types of data (images, point list, mesh, *etc.*) to be used as input data for the modules. For example, the module of primitive extraction extracts ellipses centers from images and outputs an indexed point list usable in the module of camera calibration. Only the module of videogrammetry will be discussed in detail in Section 5.1.2 but numerous other modules have been developed (see Section 5.1.1): module of camera calibration (linear and non-linear), primitive extraction (circles and crosses), two-dimensional mesh generator, strain computation, point tracking through time, *etc.* *Visiocre* uses 3 libraries (see Fig. 5.1), described in Section 5.2: *Qt* and *GUIlib* for the GUI and *v3d* for the computer vision algorithms.

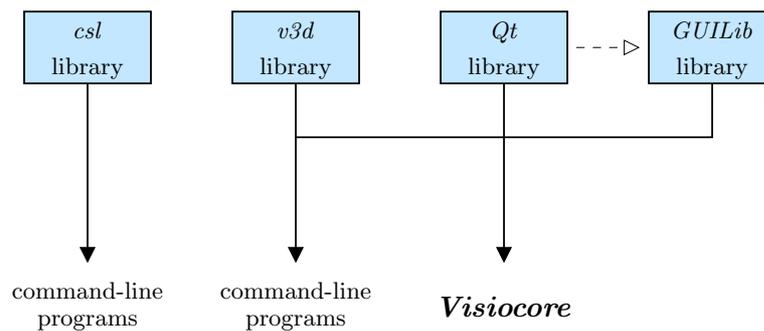


Fig. 5.1 — *Visiocre* uses 3 libraries: *Qt* and *GUIlib* for the Graphic User Interface and *v3d* for the computer vision algorithms. Besides, *v3d* and *csl* are also used to implement command-line programs.

5.1.1 Modules of Computer Vision

Visiocre is a modular software gathering a lot of different modules oriented to computer vision applications. Most of these modules are quickly described thereafter.

³We wish to acknowledge Dorian Garcia for its participation in the software architecture development.

Module of Primitive Extraction

The module of primitive extraction is useful for classical calibration using calibration target with fiducial marks. It allows to extract circles or crosses from images (see Fig. 5.2) and to return an indexed point list for the module of calibration.

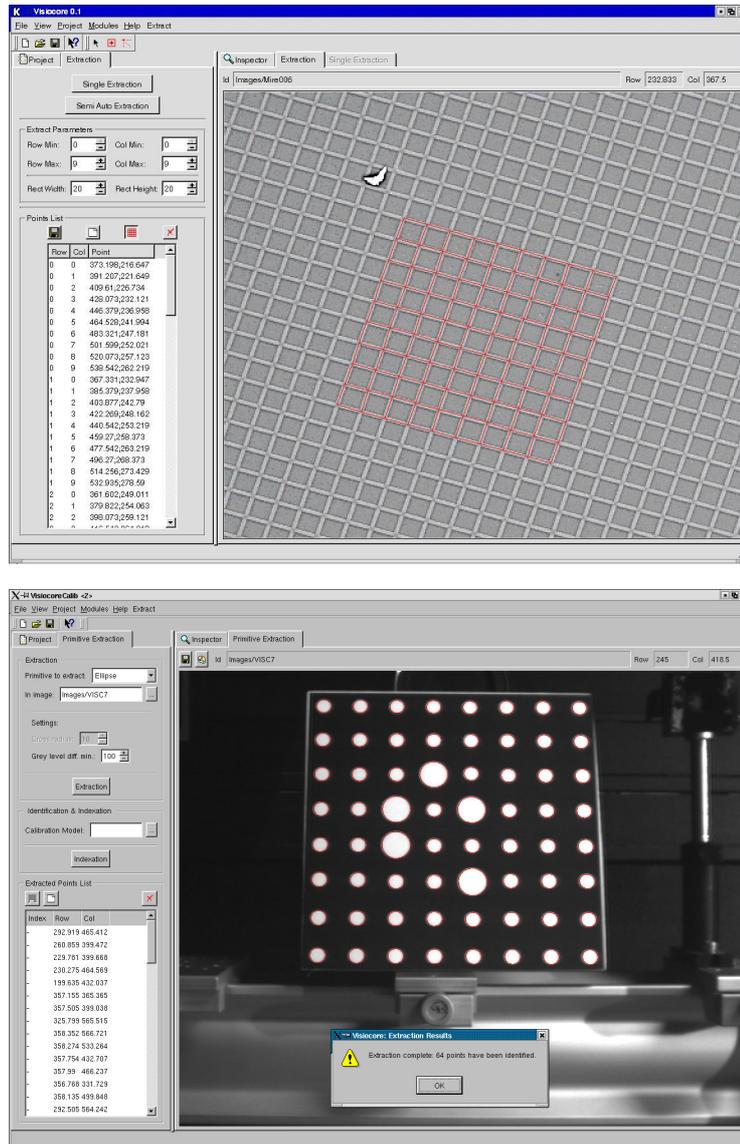


Fig. 5.2 — The module of primitive extraction allows to extract fiducial marks such as crosses or circles from calibration target images.

Module of 2D mesh

The module of 2D mesh (see Fig. 5.3) is able to mesh a set of given points using the constrained Delaunay triangulation algorithm. It is possible to impose maximum triangle areas, maximum angles, holes in the mesh, to transform edges into boundaries and *vice-versa*, etc.

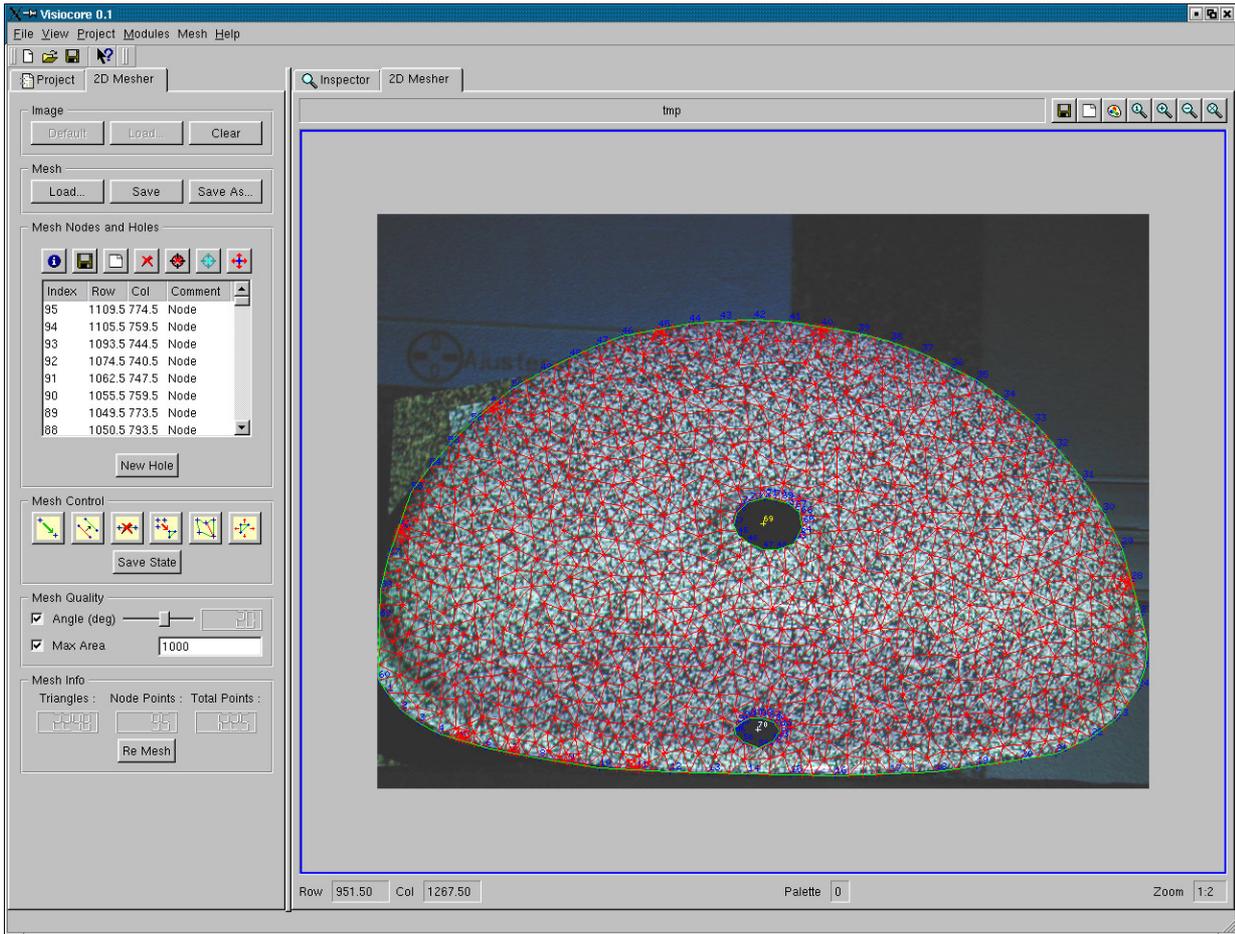


Fig. 5.3 — The module of 2D mesh realizes a constrained Delaunay triangulation from a set of given points.

Module of Tracking

The module of tracking has been realized with the help of Bruno Remondi, trainee from INSA, an engineering school of Toulouse (France). As the name suggests, this module is used to track points over an image sequence using Digital Image Correlation (see Fig. 5.4).

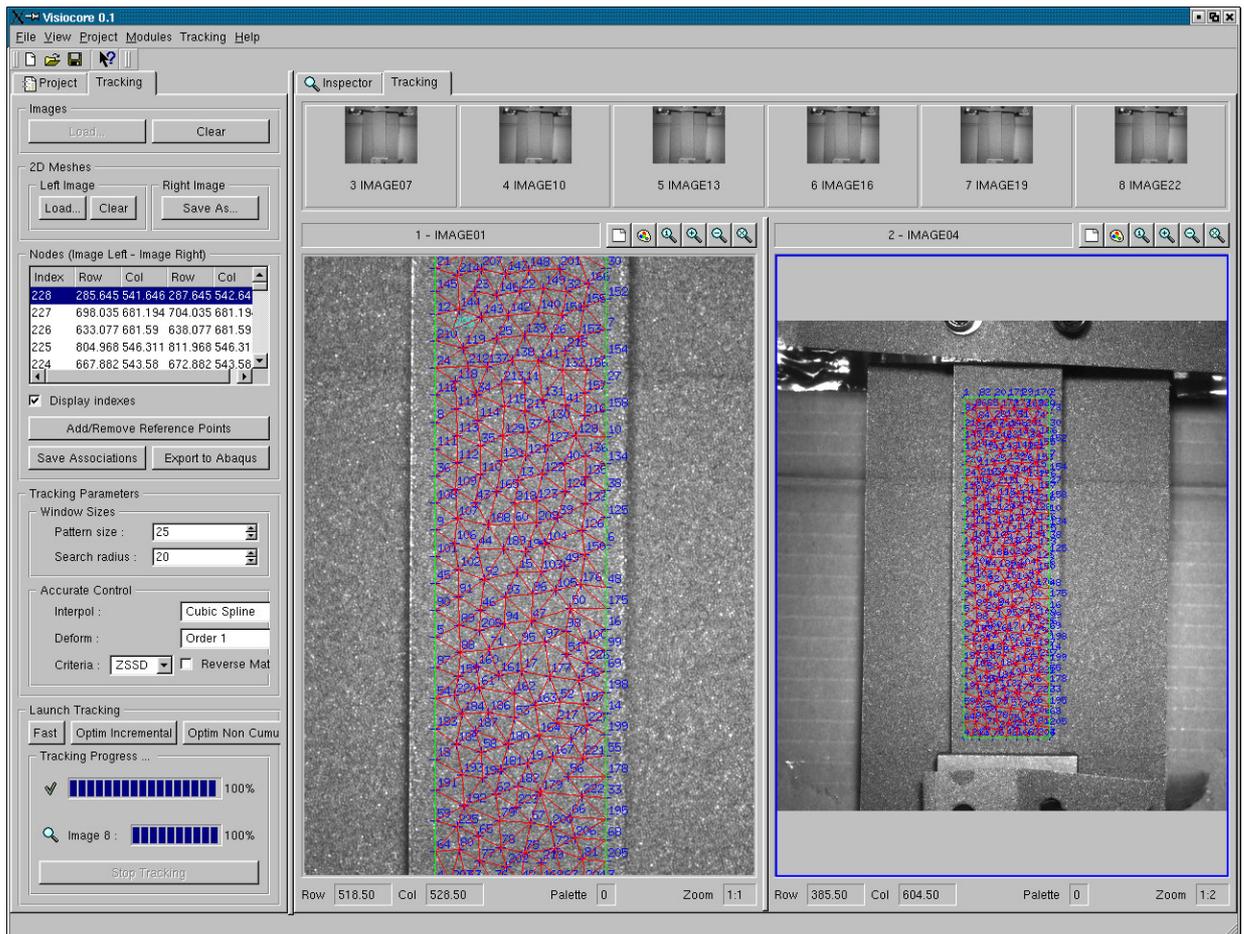


Fig. 5.4 — The module of tracking allows to track points distributed over a mesh (see module of 2D mesh, page 156) between different images of a sequence. This example presents images of a foam sample subjected to a tensile test. The natural texture of the foam allows to use Digital Image Correlation for the 2D displacement measurement.

Module of Calibration

The calibration methodology performed in this module is the traditional calibration method [Gar01], which was used before the development of our new calibration approach (which is only implemented as a command-line program). The module is able to accomplish a linear calibration [RAS93] before the non-linear optimization and takes into account the parametric distortions (radial, decentering and prismatic). In addition to the projection model parameters, the quality of the calibration is given (residue values depending on the views). It is also possible to display a distortion map and to correct images for the estimated distortion (see Fig. 5.5).

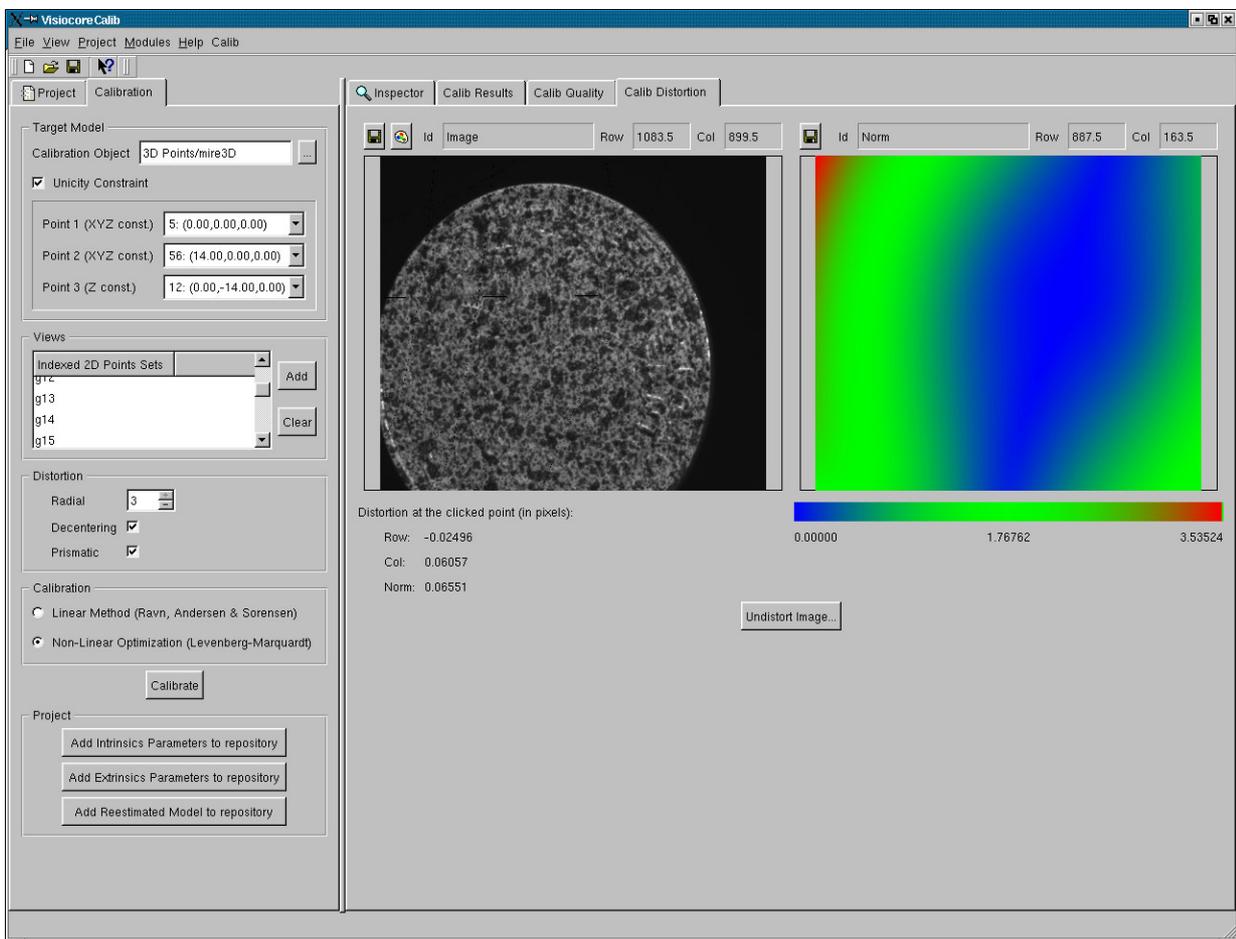


Fig. 5.5 — The module of calibration displays the results in three different tabs: the projection model parameters in the first, the calibration quality in the second and the distortion map in the last.

Module of Rigid-Body Transformations

Extrinsic parameters (rigid-body transformation in 3D space) can be represented by several ways and depending on the situation, it can be wiser to use one specific representation instead of another one. Fig. 5.6 shows the module of rigid-body transformations able to create extrinsic parameters or to convert them from one representation to another: rotation vector / rotation matrix / Euler angles + translation vector or matrix 4×4 . (see Appendix C for conversions between rotation representations).

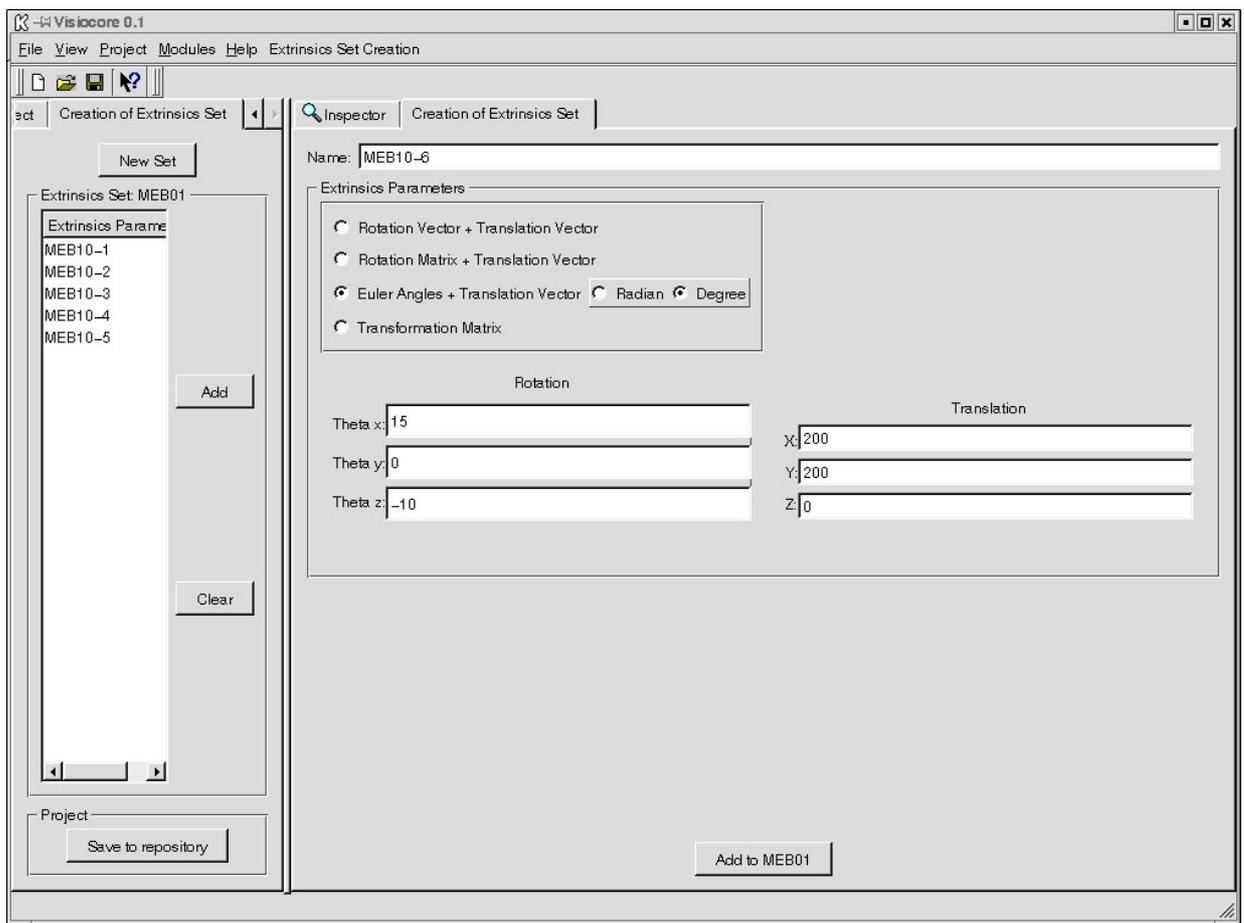


Fig. 5.6 — *The module of calibration allows to calibrate a camera imaging system .*

5.1.2 Module of Videogrammetry

The module of videogrammetry has been implemented following the algorithm described in Fig. 3.1. Calibration process is not included in the module of videogrammetry so that the module stays independent of the imaging system used (because calibration process for a SEM is not the same as the calibration process for a camera for example). Images have to be corrected for distortion in another module before being loaded and the projection model parameters are necessary inputs of the module. Most of intern parameters are adjustable and have to be correctly set up in order to obtain good results (Harris threshold, cornerness similarity, number of buckets, correlation window size, *etc.*). because even if default values are fixed, ideal values usually vary from an experiment to another.

The process is entirely automated and it allows, in the simplest case, from an image sequence to 1) extract feature points in the image, 2) match them between pairs of images, 3) estimate the epipolar geometry and the rigid-body motion between viewpoints and 4) recover the 3D shape by bundle adjustment. The 3D shape can be represented by a cloud of 3D points or a 3D mesh with or without texture mapping (see Fig. 5.8 and Fig. 5.9). In complement of the Harris detector (see Section 3.3), the GUI allows to add correspondences by clicking manually a point in an image and finding its correspondences in the entire sequence manually (user click), semi-manually (user click restricted upon the epipolar line, see Fig. 5.7) or automatically (by correlation). It is also possible to load dense disparity maps from *Vic2D* [CSI], very efficient software for dense correlation, in order to have directly an excellent estimation of the epipolar geometry.

Experimental results of 3D shape measurement using this software are presented in Section 3.5.

5.2 Libraries

For the needs of the thesis and the software, three libraries were developed: *GUIlib* (graphic library), *v3d* and *csl* (scientific libraries). Moreover, *Qt* library is used for the GUI of *Visiocre*. Fig. 5.1 illustrates the role of each library.

5.2.1 Graphic Libraries: *Qt* and *GUIlib*

Qt library is actually more than a graphic library because it includes as well container classes, input / output devices and even database and networking classes in the last versions. However, only the GUI widget classes of *Qt* are used in *Visiocre*, all the other useful classes being managed by our library *v3d*. *Qt* offers all types of basic widgets: buttons, check boxes, scrollbars, *etc.* but also some advanced widgets such as listviews, progress bars and tab dialogs.

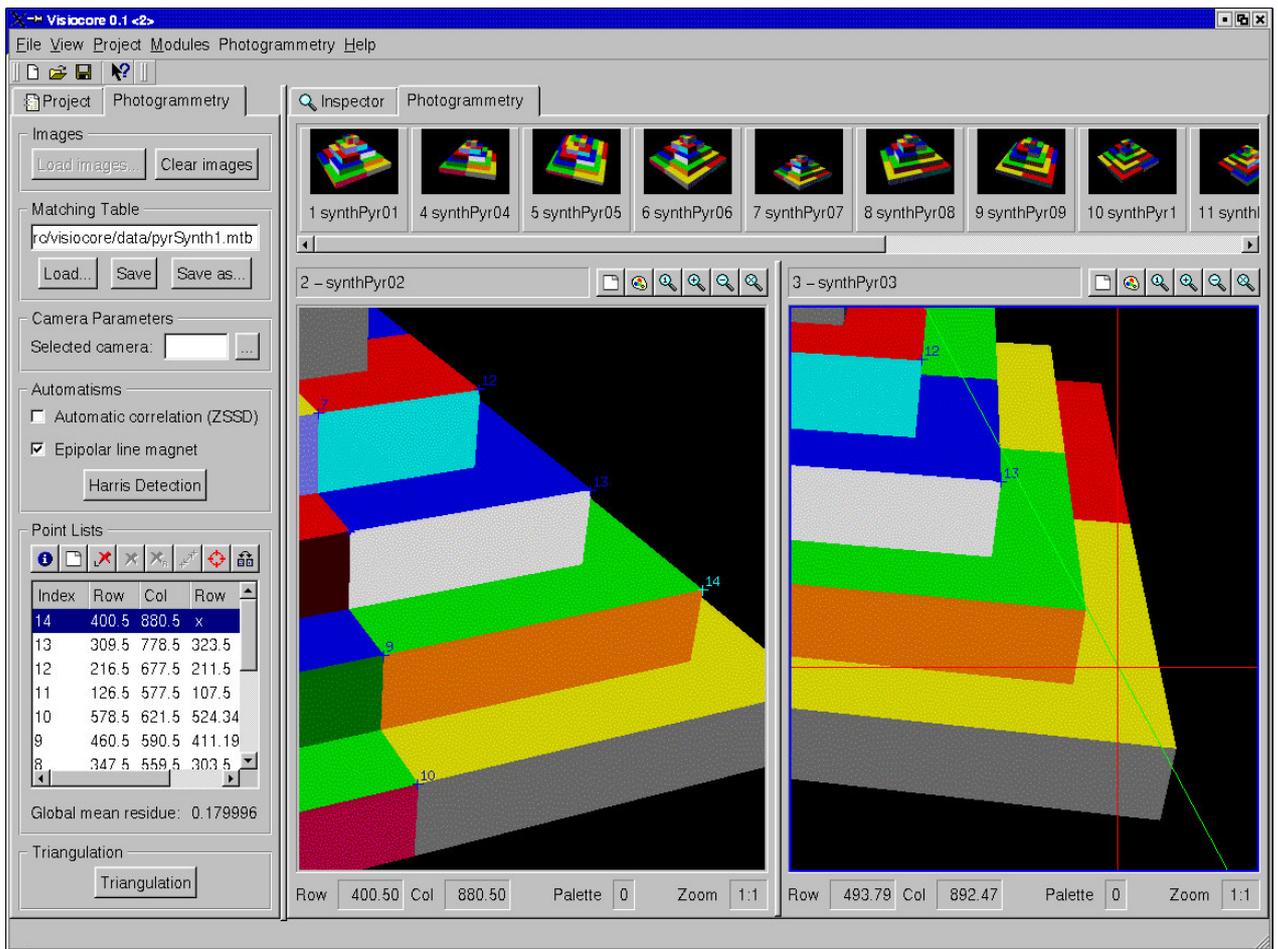


Fig. 5.7 — Screenshot of the module of videogrammetry: the epipolar geometry being estimated from the 13 first pairs of correspondences, the correspondence of the point 14 (highlighted point in the left view) must lie on its epipolar line (green line in the right view). The mouse position (represented by the crosshair) is then restricted to this line and the user can click manually the correspondence.

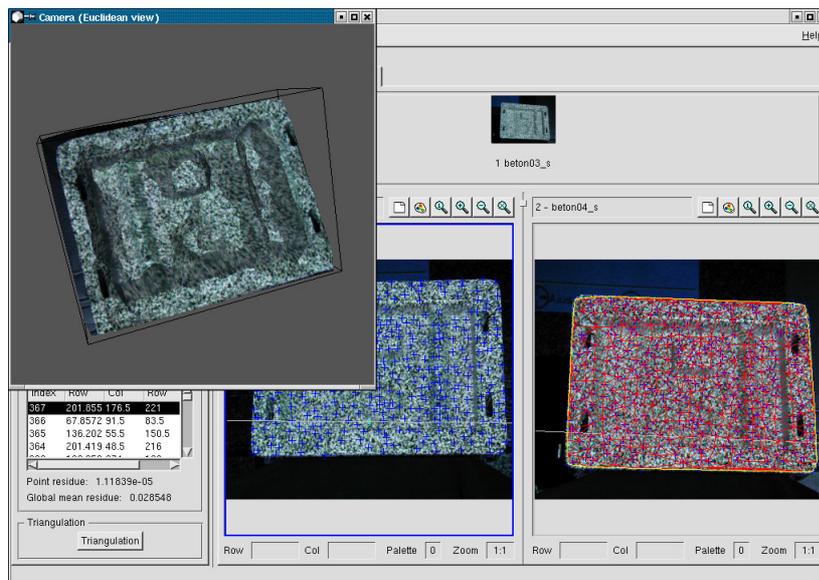


Fig. 5.8 — *Screenshot of the module of videogrammetry: 3D reconstruction of a concrete mold*

GUILib is a graphic library developed to add some graphic functionalities necessary for our applications. Here is a non-exhaustive list:

- **primitive drawing** is used mainly in the module of primitive extraction. For instance, it allows to draw an ellipse over the image from its equation, draw a grid from all the coordinates of the points extracted, *etc.* It is also employed to draw epipolar lines or correlation windows in the videogrammetry module and 2D mesh in the associated module. At last, text drawing is often used to name or enumerate sets of points in order to distinguish them.
- **crosshair drawing** instead of the mouse cursor and management of non-integer coordinates is useful to locate precisely a particular point in one image
- **restriction of mouse movement** allows to define where the mouse cursor can go. For example, the crosshair can be stuck to an epipolar line in order to manually click the correspondence of a given point in the videogrammetry module (the crosshair is projected from the actual mouse coordinates to the closest point on the epipolar line).

All the drawings are vectorial (opposed to bitmap) and are obviously on a different layer than the image. This allows to hide them, remove them, change the color, the size of them, *etc.* very easily. Using *Qt*, a custom widget of image visualization has been finally

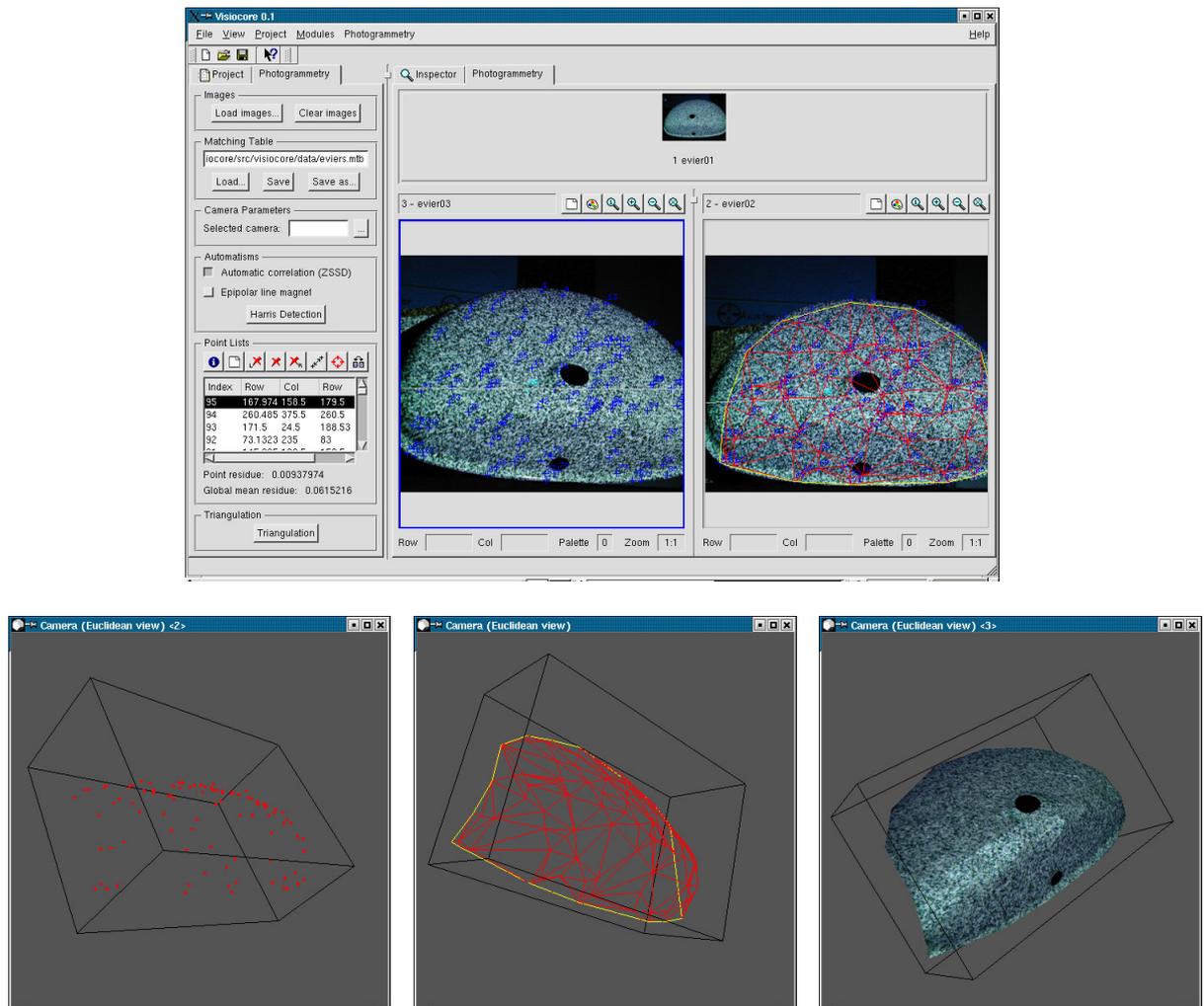


Fig. 5.9 — Screenshots of the module of videogrammetry: 3D reconstruction of a blank sink with a projected speckle pattern image on it (from left to right: cloud of points, meshed and textured reconstruction)

developed and is integrated in *Visiocre*. It allows to have all the new functionalities of *GUIlib* available at any time by a simple click (see Fig. 5.10).

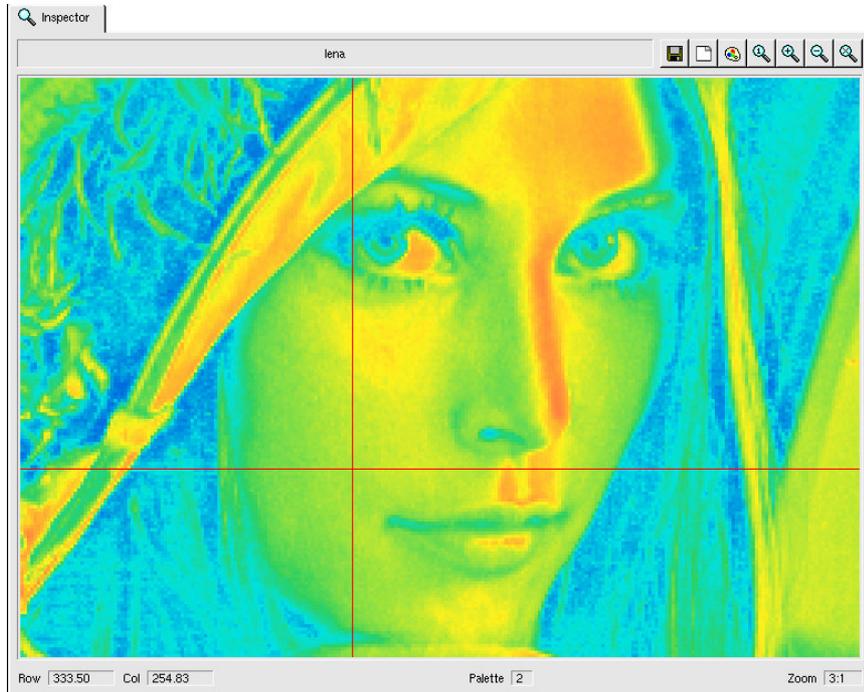


Fig. 5.10 — *The image visualization widget offers some functionalities: change of palette, zoom, crosshair drawing, non-integer coordinates, etc.*

5.2.2 Scientific Libraries: *v3d* and *csl*

v3d is the scientific library developed at Ecole des Mines d'Albi-Carmaux. Its objective is to provide all the necessary tools for computer vision applications. In this sense, it first includes useful low level mathematic classes: matrix, vector, 2D point, 3D point, rigid transformation, rotation (Euler angles, rotation vector and quaternion) are the basis of the library. Each class offers all the possible methods depending on its nature: addition, subtraction, transposition, cross product, norm, *etc.* Matrix class is a little more complex and some basic linear algebra algorithms are implemented as well: Singular Value Decomposition, Gauss-Jordan elimination, Moore-Penrose pseudo-inverse computation, *etc.*

Then, classes and algorithms purely dedicated to computer vision applications are implemented in *v3d*. They can be classified by groups:

- **calibration classes and algorithms:** linear calibration using either the method of Ravn *et al.* [RAS93] or the method of Zhang [Zha98], non linear calibration [Gar01], improved implementation of the bundle adjustment (see Appendix A), cross

and ellipse extraction (requiring image processing such as binarization), parametric distortion correction (requiring image interpolation), *etc.*

- **Digital Image Correlation classes and algorithms:** four different criteria are implemented (Sum of Squared Differences, Zero-mean SSD, Normalized Cross Correlation and Zero-mean NCC) and can be used with different methods of matching: matching along an epipolar line, interest point matching with Least Median of Squares outlier filtering (see Section 3.3.2), tracking (matching through a sequence of images), *etc.*
- **3D reconstruction / videogrammetry classes and algorithms:** fundamental matrix with all the methods described in Section 3.1 (*8-point algorithm*, isotropic scaling, rank-2 constraint enforcing, linear and non-linear estimation), essential matrix and motion recovery, projection matrix, different methods of triangulation (see Section 3.4.2), improved Harris corner detector, *etc.*

The *csf* library has been developed after *v3d* at the University of South Carolina. It provides almost the same functionalities as the *v3d* library but the implementation of all the classes and the algorithms has been optimized to be faster and memory-efficient. Moreover, it offers new classes very useful for our calibration approach such as the classes of 1D and 2D spline and their associated methods: 1D and 2D fitting, spline inversion, spline derivatives, *etc.* By lack of time, most of the applications developed with this library are only command-line programs.

Bibliography

- [AGO00] B. Aubé, D. Garcia, and J.-J. Orteu. Un scanner 3D pour la modélisation d'objets, simple à utiliser et n'utilisant qu'une seule caméra – An easy-to-use 3D scanner for object modelling using a single camera. In *SCANNING 2000: The Scanning European Meeting Point, Fifth Edition*, Paris (France), May 2000.
- [CLVWdV02] E. Cecchi, J.M. Lavest, and B. Van Wyk de Vries. Videogrammetric reconstruction applied to volcanology: perspectives for a new measurement technique in volcano monitoring. *Image Analysis and Stereology*, 21(1):31–36, 2002.
- [Cor01] Nicolas Cornille. Numérisation 3D d'objets observés au MEB (Microscope Electronique à Balayage). Master's thesis, Institut National des Sciences Appliquées de Toulouse, Jun 2001.
- [CSI] Vic2D[®] and Vic3D[®], Correlated Solutions Inc., <http://www.correlatedsolutions.com>.
- [Gar01] Dorian Garcia. *Mesure de formes et de champs de déplacements tridimensionnels par stéréo-corrélation d'images*. PhD thesis, Institut National Polytechnique de Toulouse (France), Dec 2001.
- [RAS93] O. Ravn, N. A. Andersen, and A. T. Sorensen. Auto-calibration in Automation Systems using Vision. In *3rd International Symposium on Experimental Robotics (ISER'93)*, pages 206–218, Kyoto (Japan), Oct 1993.
- [Zha98] Zhengyou Zhang. A Flexible New Technique for Camera Calibration. Technical Report MSR-TR-98-71, Microsoft Research, Dec 1998.

Conclusion & Future Work

“Accurate 3D Shape and Displacement Measurement using a Scanning Electron Microscope” is an ambitious objective because it faces two main challenges:

- Scanning electron microscope is nowadays one of the best tool to observe any phenomena at the micro- and nano-scale but it remains a visualization instrument and does not allow for the while to realize accurate two- or three-dimensional measurements. Key point of the problem is that the image formation is biased by several image disturbances and a particular attention must be paid to the modeling of the imaging process.

▷ The model of the SEM imaging system presented in this work is composed of two parts: first, a linear part corresponding to the projection and then, a non-linear part corresponding to the underlying distortions. The first part is easily established and two different models of projection have been determined depending on the magnification (see Section 2.2.2). The second part is more delicate and two complementary models of distortion are used (see Section 2.2.3): a distortion depending on the pixel location in the image (referred to as “spatially-varying distortion”) and a distortion depending on the time (referred to as “drift” or “time-varying distortion”). We developed in the thesis a new original methodology for correcting for the distortions because most of the previous works about 2D/3D measurements using a SEM consider the imaging process as a pure projection model. The few works taking into account distortion use simple parametric models which are not efficient for a complex imaging system such as the SEM and always neglect the drift effect. The calibration procedure can use any quasi-planar object with a speckle pattern texture. It leads to greater accuracy than using calibration target composed of fiducial marks because the extraction of their measure points is less accurate and their number a lot smaller. Experimental results are presented (see Section 2.4) to compare the different approaches and demonstrate the great accuracy improvement realized with our method, not only with the SEM but also with other imaging systems: compared to a calibration neglecting distortions, standard deviation of the magnitudes of reprojection errors using our calibration methodology is improved from 0.43 to 0.02 pixels for

the SEM imaging system, from 0.08 to 0.005 pixels for a stereo-optical microscope and from 0.1 to 0.03 pixels for a camera.

- The Scanning Electron Microscope provides only a single sensor whereas 3D shape and displacement measurement usually requires two imaging system for recovering 3D information using the classical stereo-vision approach.

▷ The specimen being mounted on the mobile SEM stage, images can be acquired from multiple viewpoints under the hypothesis that the specimen shape does not change during the time of acquisition of the images. 3D shape and displacement measurement is then possible using the principle of videogrammetry for recovering the unknown rigid-body motions undergone by the specimen (see Section 3). Moreover, multi-view imaging allows not only to obtain a better accuracy in the 3D reconstruction but also to have a complete view around the observed specimen. This approach is not specific to reduced-length scale applications and in addition to experimental results using the SEM, an experiment using a single camera is presented (see Section 3.5). This same experiment was also the occasion to compare the results with the data obtained by a 3D laser scanner and by this way, to assess the accuracy of the developed method (mean distance between the two clouds of points: 0.012 mm). For the needs of this work, a modular software gathering different computer vision applications (calibration, primitive extraction, point tracking, strains computation, *etc.*) and particularly a module of videogrammetry has been developed (see Section 5).

In conclusion, a measurement methodology has successfully been developed for analyzing and correcting SEM images so that quantitative 3D shape and displacement measurements can be obtained. However, further investigations, mainly experimental, are necessary to validate the work presented in this dissertation.

Regarding the calibration and distortion correction, initial experimental results are very promising but they have to be confirmed by more experiments. In particular, new high magnification experiments are indispensable to definitely corroborate the proposed approach for drift correction. Moreover, it would be also interesting to use an Environmental Scanning Electron Microscope and analyze the influence of the low-vacuum mode (environmental mode allowing to visualize non-metallic specimen) on the spatially-varying and time-varying distortions by comparison with a SEM in high-vacuum mode.

Regarding the 3D measurements, more experiments are also required. Indeed, the accuracy of the 3D shape measurement has been validated at the macro-scale (see Section 3.5.2) but the equivalent experiment at reduced-length scale is still necessary. For example, a comparison can be made between the 2 clouds of 3D points obtained on the one hand with a SEM and the videogrammetry technique and on the other hand with an AFM (Atomic Force Microscope). A second experiment that can be intended would be

the 3D reconstruction of a specimen of which the dimensions are perfectly known. For the 3D displacement and strain measurements, work is on-going to conduct experiments with a specimen undergoing tensile loading in a specially designed SEM loading stage and comparison with strain gauges will assess the accuracy of the developed method.

Appendix A

Bundle adjustment Implementation

This appendix¹ presents an efficient way to implement the bundle adjustment procedure taking into account the specific matrix structure of the equation system to solve. This section considers that the reader already acquired the basic knowledge about bundle adjustment (for example, Levenberg-Marquardt or Newton iteration and Jacobian and Hessian matrices will not be introduced here and the reader may first refer to the Triggs *et al.*'s synthesis of bundle adjustment [TMHF00]). The matrix presentation used thereafter is inspired from Hartley [Har94] and has already been used in previous works [Lav96, Pol00].

Let's consider the more complete case of bundle adjustment where we take a perspective model for the imaging system and a parametric distortion included in the optimization process. We remind the objective function to minimize (see Equ. (2.22)):

$$\min_{\mathbf{M}_i, \mathbf{T}^j, \mathbf{K}, \mathbf{d}} \sum_i \sum_j \|\mathbf{m}_i^j - \mathbf{n}_i^j\|_2^2 \quad (\text{A.1})$$

where \mathbf{M}_i are the 3D model points, \mathbf{T}^j the rigid-body transformations between the views (extrinsic parameters), \mathbf{K} the intrinsic parameter matrix and $\mathbf{d} = (k_1 \ k_2 \ k_3 \ d_1 \ d_2 \ p_1 \ p_2)$ is the parameter vector of distortion.

Let's consider a standard calibration experiment composed of $n = 10$ views of the calibration target where we take $p = 3000$ points. The number of available equations is $2np = 60000$ equations (one equation per coordinate of 2D projected points). The parameters to estimate are: 5 intrinsic parameters $\mathbf{k} = (c_x \ c_y \ f_x \ f_y \ s)$, 7 distortion parameters, 6 extrinsic parameters per view (n_x, n_y, n_z, t_x, t_y et t_z)² and 3 coordinates per 3D model point. We saw in Section 2.3.1 that 7 parameters remain fixed to enforce the uniqueness of the solution. The total number of parameters to estimate is then $5 + 7 + 6n + 3p - 7 = 9065$. Such a minimization would be very computationally expensive

¹We would like to thank Dorian Garcia for his participation in the work presented in this appendix.

²The rotation is here represented by a rotation vector of parameters n_x, n_y and n_z (see Appendix C) because it is a minimal representation, unlike the rotation matrix.

(several hours with a brand new computer) if the normal equations are solved directly and we propose to reduce this time to a few minutes by taking into account the very specific structure of the implicated matrices. As illustrated in Fig. A.1, the Jacobian matrix \mathbf{J} (first derivative of the objective function useful for the Levenberg-Marquardt algorithm) is very sparse and has a block structure.

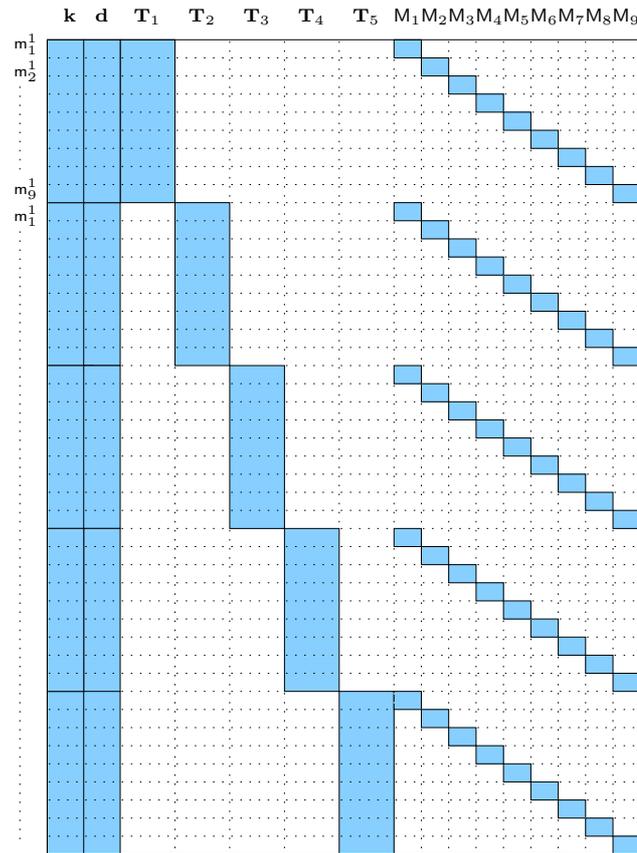


Fig. A.1 — *Sparse structure of the Jacobian matrix \mathbf{J} for a calibration composed of five views and ten 3D model points. Null elements of the matrix are represented by blank parts.*

In addition to the Jacobian matrix, normal equations to solve and Levenberg-Marquardt algorithm require the Hessian matrix \mathbf{H} (second derivative of the objective function). It is commonly approximated as $\mathbf{H} = \mathbf{J}^\top \mathbf{J}$ and is therefore a symmetric positive definite matrix. The block and sparse structure of \mathbf{H} is represented in Fig. A.2.

Minimizing the objective function (A.1) using the Levenberg-Marquardt or the Newton iteration consists in approximating progressively δ such that $\mathbf{H} \delta = \epsilon$ (called normal equation) where ϵ is the residue vector. The standard implementation of the bundle adjustment inverts directly the matrix \mathbf{H} at each iteration but because the size of this matrix is generally huge (9065×9065 in the previous example), it is very time-consuming and the

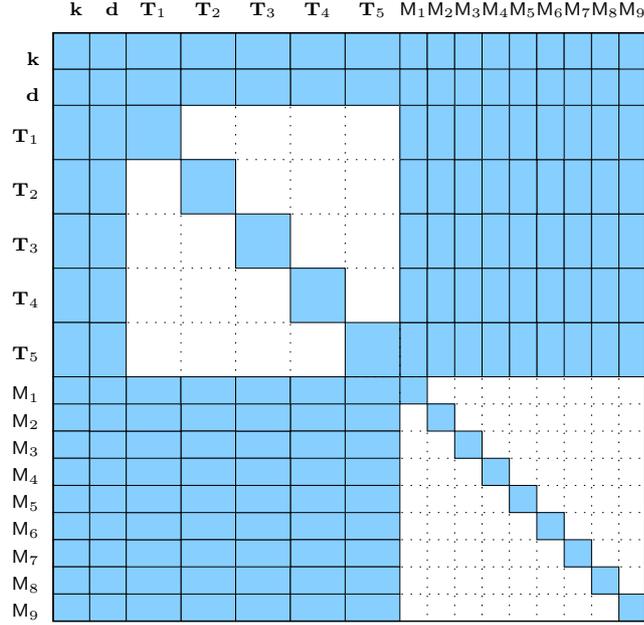


Fig. A.2 — *Approximated Hessian matrix structure: \mathbf{H} has a block and sparse structure that will be used for efficient implementation of the bundle adjustment procedure.*

whole process can last several hours. A first possible improvement to invert \mathbf{H} would be to implement a specific algorithm of symmetric positive definite matrix inversion such as Cholesky's or Rutishauser's but the most efficient implementation, presented in this section and allowing to reduce considerably the computation time, consists in inverting \mathbf{H} using its block structure. Fig. A.3 shows the way we choose to split \mathbf{H} in order to simplify the computations by explicitly formulating each block \mathbf{H}_{11} , \mathbf{H}_{12} and \mathbf{H}_{22} .

The partial derivatives of the projected point with respect to the intrinsic parameters, the distortion parameters, the extrinsic parameters and the 3D model points are expressed as follows:

$$\begin{aligned} \frac{\partial \mathbf{m}_i^j}{\partial \mathbf{k}} &= \frac{\partial \mathbf{m}}{\partial \mathbf{k}}(\mathbf{k}, \mathbf{d}, \mathbf{T}_j, M_i) = \begin{pmatrix} \frac{\partial x}{\partial c_x} & \frac{\partial x}{\partial c_y} & \frac{\partial x}{\partial f_x} & \frac{\partial x}{\partial f_y} & \frac{\partial x}{\partial s} \\ \frac{\partial y}{\partial c_x} & \frac{\partial y}{\partial c_y} & \frac{\partial y}{\partial f_x} & \frac{\partial y}{\partial f_y} & \frac{\partial y}{\partial s} \end{pmatrix} \\ \frac{\partial \mathbf{m}_i^j}{\partial \mathbf{d}} &= \frac{\partial \mathbf{m}}{\partial \mathbf{d}}(\mathbf{k}, \mathbf{d}, \mathbf{T}_j, M_i) = \begin{pmatrix} \frac{\partial x}{\partial k_{k_1}} & \frac{\partial x}{\partial k_{k_2}} & \dots \\ \frac{\partial y}{\partial k_{k_1}} & \frac{\partial y}{\partial k_{k_2}} & \dots \end{pmatrix} \\ \frac{\partial \mathbf{m}_i^j}{\partial \mathbf{T}_j} &= \frac{\partial \mathbf{m}}{\partial \mathbf{T}}(\mathbf{k}, \mathbf{d}, \mathbf{T}_j, M_i) = \begin{pmatrix} \frac{\partial x}{\partial n_x} & \frac{\partial x}{\partial n_y} & \frac{\partial x}{\partial n_z} & \frac{\partial x}{\partial t_x} & \frac{\partial x}{\partial t_y} & \frac{\partial x}{\partial t_z} \\ \frac{\partial y}{\partial n_x} & \frac{\partial y}{\partial n_y} & \frac{\partial y}{\partial n_z} & \frac{\partial y}{\partial t_x} & \frac{\partial y}{\partial t_y} & \frac{\partial y}{\partial t_z} \end{pmatrix} \\ \frac{\partial \mathbf{m}_i^j}{\partial M_i} &= \frac{\partial \mathbf{m}}{\partial M}(\mathbf{k}, \mathbf{d}, \mathbf{T}_j, M_i) = \begin{pmatrix} \frac{\partial x}{\partial X} & \frac{\partial x}{\partial Y} & \frac{\partial x}{\partial Z} \\ \frac{\partial y}{\partial X} & \frac{\partial y}{\partial Y} & \frac{\partial y}{\partial Z} \end{pmatrix} \end{aligned}$$

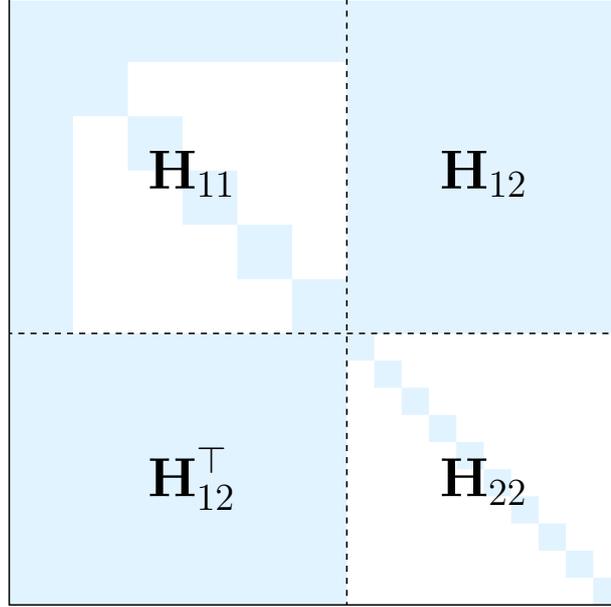


Fig. A.3 — *The approximate Hessian matrix \mathbf{H} is split into the block matrices \mathbf{H}_{11} , \mathbf{H}_{12} and \mathbf{H}_{22} . Since \mathbf{H} is a symmetric positive definite matrix, \mathbf{H}_{11} and \mathbf{H}_{22} are symmetric positive definite as well.*

For convenience, let's note $\boldsymbol{\kappa} = \begin{pmatrix} \mathbf{k} & | & \mathbf{d} \end{pmatrix}$ the combination of the intrinsic and distortion parameters. The partial derivatives of the projected point with respect to this new vector are $\frac{\partial \mathbf{m}_i^j}{\partial \boldsymbol{\kappa}} = \frac{\partial \mathbf{m}}{\partial \boldsymbol{\kappa}}(\mathbf{k}, \mathbf{d}, \mathbf{T}_j, \mathbf{M}_i) = \begin{pmatrix} \frac{\partial \mathbf{m}_i^j}{\partial \mathbf{k}} & | & \frac{\partial \mathbf{m}_i^j}{\partial \mathbf{d}} \end{pmatrix}$. Moreover, block matrices \mathbf{H}_{11} , \mathbf{H}_{12} and \mathbf{H}_{22} are split in new block matrices represented in Fig. A.4 and formulated as follows:

$$\begin{aligned} \mathbf{A}^{12 \times 12} &= \sum_{i=1}^n \sum_{j=1}^p \left(\frac{\partial \mathbf{m}_{ij}}{\partial \boldsymbol{\kappa}} \right)^\top \frac{\partial \mathbf{m}_{ij}}{\partial \boldsymbol{\kappa}} \\ \mathbf{B}_j^{12 \times 6} &= \sum_{i=1}^p \left(\frac{\partial \mathbf{m}_{ij}}{\partial \boldsymbol{\kappa}} \right)^\top \frac{\partial \mathbf{m}_{ij}}{\partial \mathbf{T}_i} \\ \mathbf{C}_i^{6 \times 6} &= \sum_{j=1}^p \left(\frac{\partial \mathbf{m}_{ij}}{\partial \mathbf{T}_i} \right)^\top \frac{\partial \mathbf{m}_{ij}}{\partial \mathbf{T}_i} \\ \mathbf{D}_j^{12 \times 3} &= \sum_{i=1}^n \left(\frac{\partial \mathbf{m}_{ij}}{\partial \boldsymbol{\kappa}} \right)^\top \frac{\partial \mathbf{m}_{ij}}{\partial \mathbf{M}_j} \\ \mathbf{E}_{ij}^{6 \times 3} &= \left(\frac{\partial \mathbf{m}_{ij}}{\partial \mathbf{T}_i} \right)^\top \frac{\partial \mathbf{m}_{ij}}{\partial \mathbf{M}_j} \\ \mathbf{F}_j^{3 \times 3} &= \sum_{i=1}^n \left(\frac{\partial \mathbf{m}_{ij}}{\partial \mathbf{M}_j} \right)^\top \frac{\partial \mathbf{m}_{ij}}{\partial \mathbf{M}_j} \end{aligned}$$

Normal equations to solve become:

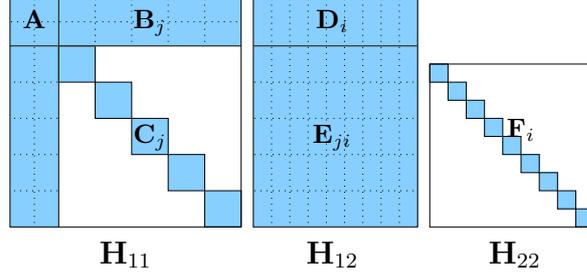


Fig. A.4 — *Composition of block matrices of the Hessian matrix: \mathbf{H}_{11} , \mathbf{H}_{12} and \mathbf{H}_{22} are split in new block matrices for convenience.*

$$\begin{pmatrix} \mathbf{H}_{11} & \mathbf{H}_{12} \\ \mathbf{H}_{12}^\top & \mathbf{H}_{22} \end{pmatrix} \begin{pmatrix} \delta_1 \\ \delta_2 \end{pmatrix} = \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \end{pmatrix}$$

By multiplying the two sides of the equality by the following matrix:

$$\begin{pmatrix} \mathbf{I} & -\mathbf{H}_{12}\mathbf{H}_{22}^{-1} \\ \mathbf{0} & \mathbf{I} \end{pmatrix}$$

we obtain the new system to solve:

$$\begin{pmatrix} \mathbf{H}_{11} - \mathbf{H}_{12}\mathbf{H}_{22}^{-1}\mathbf{H}_{12}^\top & \mathbf{0} \\ \mathbf{H}_{12}^\top & \mathbf{H}_{22} \end{pmatrix} \begin{pmatrix} \delta_1 \\ \delta_2 \end{pmatrix} = \begin{pmatrix} \epsilon_1 - \mathbf{H}_{12}\mathbf{H}_{22}^{-1}\epsilon_2 \\ \epsilon_2 \end{pmatrix} \quad (\text{A.2})$$

Let's note $\mathbf{\Lambda} = \mathbf{H}_{11} - \mathbf{H}_{12}\mathbf{H}_{22}^{-1}\mathbf{H}_{12}^\top$. This computation will not be made directly but using the block structure of \mathbf{H} in order to be faster. As presented in Fig. A.5, the structure of $\mathbf{\Lambda}$ is $\begin{pmatrix} \mathbf{U} & \mathbf{V} \\ \mathbf{V}^\top & \mathbf{W} \end{pmatrix}$ with:

$$\begin{aligned} \mathbf{U}^{12 \times 12} &= \mathbf{A} - \sum_{j=1}^p \mathbf{D}_j \mathbf{F}_j^{-1} \mathbf{D}_j^\top \\ \mathbf{V}_i^{12 \times 6} &= \mathbf{B}_i - \sum_{j=1}^p \mathbf{D}_j \mathbf{F}_j^{-1} \mathbf{E}_{ij}^\top \\ \mathbf{W}_{ki}^{6 \times 6} &= \delta_{ki} \mathbf{C}_i - \sum_{j=0}^p \mathbf{E}_{kj} \mathbf{F}_j^{-1} \mathbf{E}_{ij}^\top \end{aligned}$$

where δ_{ki} is the Kronecker delta (if $k = i$ then $\delta_{ki} = 1$ else $\delta_{ki} = 0$).

The system (A.2) can be solved in two steps. The first one consists in solving the following equation:

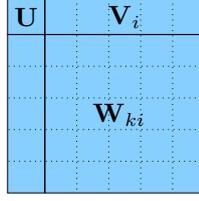


Fig. A.5 — *Structure of the matrix Λ*

$$\begin{aligned} \text{Equ. (A.2)} &\Rightarrow \underbrace{(\mathbf{H}_{11} - \mathbf{H}_{12}\mathbf{H}_{22}^{-1}\mathbf{H}_{12}^\top)}_{\Lambda} \boldsymbol{\delta}_1 = \boldsymbol{\epsilon}_1 - \mathbf{H}_{12}\mathbf{H}_{22}^{-1}\boldsymbol{\epsilon}_2 \\ &\Rightarrow \boldsymbol{\delta}_1 = \Lambda^{-1} (\boldsymbol{\epsilon}_1 - \mathbf{H}_{12}\mathbf{H}_{22}^{-1}\boldsymbol{\epsilon}_2) \end{aligned}$$

The inversion of the matrix \mathbf{H}_{22} is immediate because \mathbf{H}_{22} consists only of a set of 3×3 sub-blocks \mathbf{F}_i on its diagonal (see Fig. A.3) and \mathbf{H}_{22}^{-1} is then composed of the known inverse matrices \mathbf{F}_i^{-1} on its diagonal. Indeed, these matrices have already been computed for determining the matrices \mathbf{U} , \mathbf{V}_i or \mathbf{W}_{ki} . Moreover, each \mathbf{F}_i is symmetric positive definite and the inverse \mathbf{F}_i was determined easily:

$$\mathbf{F}_i = \begin{pmatrix} a & b & c \\ b & d & e \\ c & e & f \end{pmatrix} \Rightarrow \mathbf{F}_i^{-1} = \frac{1}{adf + 2bce - ae^2 - b^2f - c^2d} \begin{pmatrix} df - e^2 & ec - bf & be - cd \\ ec - bf & af - c^2 & bc - ae \\ be - cd & bc - ae & ad - b^2 \end{pmatrix}$$

The inversion of the matrix Λ can be realized very efficiently as well using the fact that Λ is symmetric positive definite or better, by using the Schur complement method. Indeed, block matrices can be easily inverted using the general following formula:

$$\begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{pmatrix}^{-1} = \begin{pmatrix} \mathbf{A}^{-1} + \mathbf{A}^{-1}\mathbf{B}\mathbf{S}^{-1}\mathbf{C}\mathbf{A}^{-1} & -\mathbf{A}^{-1}\mathbf{B}\mathbf{S}^{-1} \\ -\mathbf{S}^{-1}\mathbf{C}\mathbf{A}^{-1} & \mathbf{S}^{-1} \end{pmatrix}$$

where the Schur complement is $\mathbf{S} = \mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B}$.

If we apply this formula to Λ , we obtain:

$$\Lambda^{-1} = \begin{pmatrix} \mathbf{U} & \mathbf{V} \\ \mathbf{V}^\top & \mathbf{W} \end{pmatrix}^{-1} = \begin{pmatrix} \mathbf{U}^{-1} + \mathbf{X}\mathbf{V}^\top\mathbf{U}^{-1} & -\mathbf{X} \\ -\mathbf{X}^\top & \mathbf{S}^{-1} \end{pmatrix}$$

with:

$$\begin{aligned} \mathbf{S} &= \mathbf{W} - \mathbf{V}^\top\mathbf{U}^{-1}\mathbf{V} \\ \mathbf{X} &= \mathbf{U}^{-1}\mathbf{V}\mathbf{S}^{-1} \end{aligned}$$

Finally, when $\boldsymbol{\delta}_1$ is determined, the second step to compute $\boldsymbol{\delta}_2$ and solve the normal equations is:

$$\begin{aligned}\text{Equ. (A.2)} &\Rightarrow \mathbf{H}_{12}^\top \boldsymbol{\delta}_1 + \mathbf{H}_{22} \boldsymbol{\delta}_2 = \boldsymbol{\epsilon}_2 \\ &\Rightarrow \boldsymbol{\delta}_2 = \mathbf{H}_{22}^{-1} (\boldsymbol{\epsilon}_2 - \mathbf{H}_{12}^\top \boldsymbol{\delta}_1)\end{aligned}$$

Again, the computation is achieved very quickly because the inverse of \mathbf{H}_{22} has already been computed previously.

When $\boldsymbol{\delta}_1$ and $\boldsymbol{\delta}_2$ are at last known, one iteration of the Levenberg-Marquardt or the Newton algorithm can be processed to converge towards the bundle adjustment solution. Thanks to the implementation described in this section which uses the sparse and block structure of the Hessian matrix, the time of processing is greatly reduced compared to a standard implementation where the Hessian matrix is directly inverted (a few minutes instead of several hours).

Bibliography

- [Har94] Richard I. Hartley. Lines and Points in Three Views - An Integrated Approach. In *ARPA Image Understanding Workshop*, 1994.
- [Lav96] Stéphane Laveau. *Géométrie d'un système de n caméras. Théorie, estimation et applications*. PhD thesis, Ecole Polytechnique, May 1996.
- [Pol00] Marc Pollefeys. Tutorial on 3D Modeling from Images. In *European Conference on Computer Vision (ECCV'2000)*, Dublin (Ireland), Jun 2000.
- [TMHF00] Bill Triggs, Philip F. McLauchlan, Richard I. Hartley, and Andrew W. Fitzgibbon. Bundle Adjustment – A Modern Synthesis. In Springer Verlag, editor, *Vision Algorithms: Theory and Practice*, volume 1883, pages 298–372, 2000.

Appendix B

Introduction to Splines

Spline functions are functions defined piecewise by polynomial. Due to the simplicity of their representation, they are widely used in computer graphics for curve and surface fitting (interpolation or approximation)¹ and particularly for modeling complex functions. In this thesis (see Section 2.2.3), the time-varying distortion is modeled using spline curves (2D representation) and the spatially-varying distortion is modeled using spline surfaces (3D representation).

Note that this appendix is only an introduction to spline functions providing basic information in order to understand the use of spline functions in our work. The curve and surface fitting by approximation have already been developed in the dissertation (see Section 2.3.1) and the expressions of the basis functions and their derivatives depending on the degree of spline are given in Section 4.1.2, Table 4.1. The reader may also refer to [Far93, Nie98] for a more complete overview about splines.

B.1 Two-dimensional Case: Spline Curve

B.1.1 Spline Definition

Given $m + 1$ points t_0, \dots, t_m (called knots) in an interval $[a, b]$ such that $a = t_0 \leq t_1 \leq \dots \leq t_{m-1} \leq t_m = b$, the function \mathcal{S} is a spline of degree p if:

- \mathcal{S} and its first $p - 1$ derivatives are continuous in the interval $[a, b]$:

$$\mathcal{S} \in \mathbf{C}^{p-1}(a, b)$$

¹The difference between interpolation and approximation is that interpolation provides a model which passes exactly through all the data points whereas approximation minimizes a function in order to provide a model which passes the closest possible to all the data points in some sense. For example, minimizing the l_2 norm leads to an approximation in the least-squares sense, minimizing the l_∞ norm leads to an approximation in the min-max sense, *etc.*

- \mathcal{S} is a polynomial of degree at most p in the m subintervals (called knot spans) $[t_i, t_{i+1}[$ with $i = 0 \dots m - 1$:

$$\mathcal{S}_{[t_i, t_{i+1}[} \in \Pi^p, i = 0 \dots m - 1$$

Depending on the knot distribution within $[a, b]$ (see Fig. B.1), the spline is said:

- **uniform**: the knots are equidistantly distributed $t_1 - t_0 = t_2 - t_1 = \dots = t_m - t_{m-1}$
- **open uniform**: the $p + 1$ first knots are all equal $t_0 = t_1 = \dots = t_p$ and the $p + 1$ last knots as well $t_{m-p} = t_{m-p+1} = \dots = t_m$. The other knots (from t_{p+1} to t_{m-p-1}) are called interior knots and are equidistantly distributed.
- **non-uniform**: the knots are only subject to the general constraint $t_i \leq t_{i+1}$

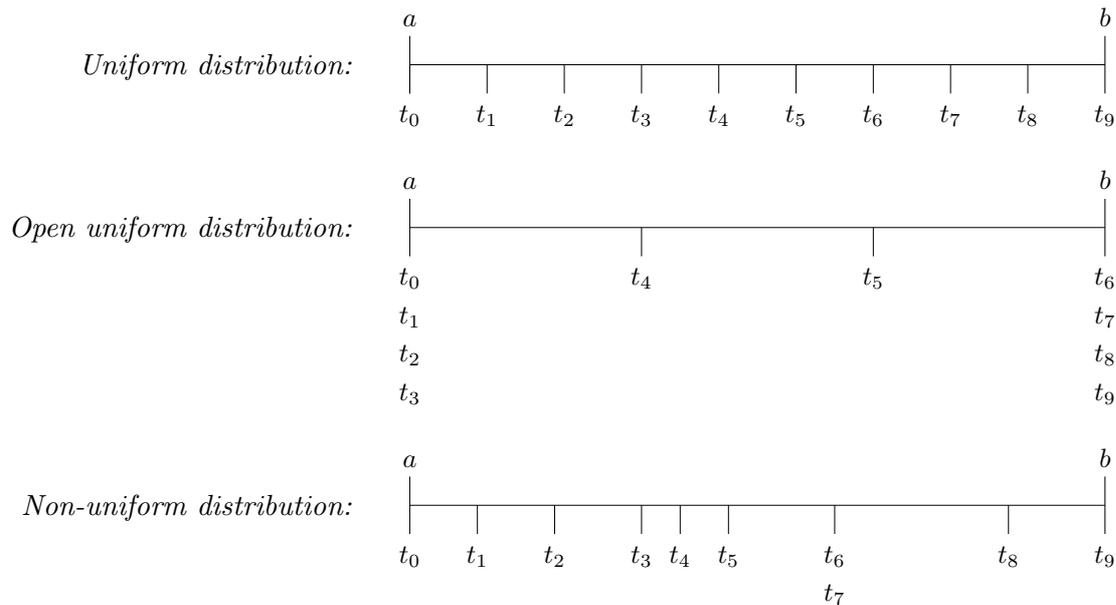


Fig. B.1 — Example of the three possible knot distribution for a cubic B-spline (degree $p = 3$) with $m + 1 = 10$ knots. From top to bottom: uniform, open-uniform and non-uniform distribution.

B.1.2 B-Spline Definition

Splines used in our work are actually uniform B-splines (short for basis-splines), a special case of spline. B-spline functions are spline functions defined such that:

$$\mathcal{S}(t) = \sum_{i=0}^n \alpha_i N_{i,p}(t)$$

where:

- $n = m - p - 1$ with $m + 1$, the number of knots and p , the degree of the B-spline
- α_i are the $(n + 1)$ spline coefficients (also called control points or de Boor points)
- $N_{i,p}$ are the basis functions (also called kernels) of the B-spline, defined recursively as follows:

$$\begin{aligned} N_{i,0}(t) &= \begin{cases} 1 & \text{if } t \in [t_i, t_{i+1}[\\ 0 & \text{otherwise} \end{cases} \\ N_{i,p \neq 0}(t) &= \frac{t - t_i}{t_{i+p} - t_i} N_{i,p-1}(t) + \frac{t_{i+p+1} - t}{t_{i+p+1} - t_{i+1}} N_{i+1,p-1}(t) \end{aligned} \quad (\text{B.1})$$

Equ. (B.1) shows that the shape of the basis functions are determined by the relative spacing between the knots. In our work, the B-splines are uniform and Fig. B.2 illustrates the shapes of the basis functions in this case.

Each point of the B-spline curve is influenced by $p + 1$ basis functions $N_{i,p}$ weighted by their associated coefficients α_i . The B-spline curve is then not available for the p first and p last knots (see Fig. B.2) and is composed of $n - p + 1$ curves joined \mathbf{C}^{p-1} continuously at the interior knots.

B.2 Three-dimensional Case: Spline Surface

A B-spline surface is defined as the tensor product of two B-splines curves:

$$\mathcal{S}(x, y) = \sum_{i=0}^{n_i} \sum_{j=0}^{n_j} \alpha_{i,j} N_{i,p}(x) N_{j,p}(y)$$

where:

- $n_i = m_i - p - 1$ and $n_j = m_j - p - 1$ with $m_i + 1$ (respectively $m_j + 1$), the number of horizontal (respectively vertical) knots and p , the degree of the B-spline
- $\alpha_{i,j}$ are the $(n_i + 1) \cdot (n_j + 1)$ spline coefficients
- $N_{i,p}$ (respectively $N_{j,p}$) are the basis functions of the B-spline along x -axis (respectively along y -axis)

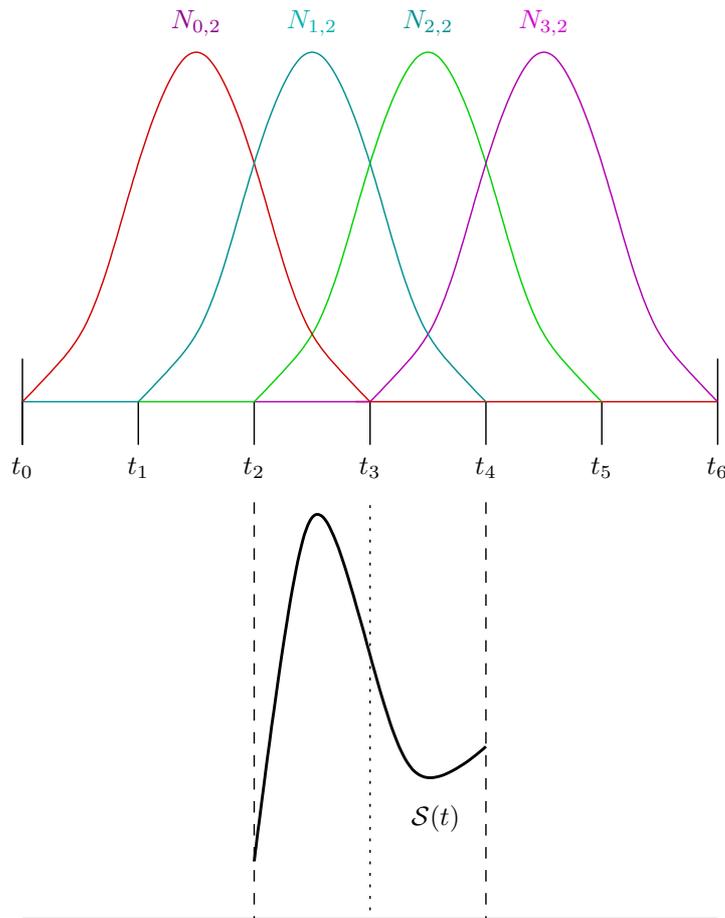


Fig. B.2 — Example of basis functions of a uniform quadratic B-spline (degree $p = 2$) with $m + 1 = 7$ knots. The B-spline is defined by $n + 1 = m - p = 4$ control points and is composed of $n - p + 1 = 2$ curves.

Similarly to the B-spline curve, the points of a B-spline surface are affected by $p+1 \times p+1$ basis functions and their coefficients. The surface is only available for the horizontal and vertical interior knots (see Fig. B.3). The rectangular area defined by an interior horizontal and vertical knot span is called a patch.

Bibliography

- [Far93] G. Farin. *Curves and Surfaces for Computer Aided Geometric Design – A Practical Guide*. Academic Press, 1993.
- [Nie98] H. B. Nielsen. Cubic splines. Lecture notes, Department of Mathematical Modeling, Technical University of Denmark, 1998.

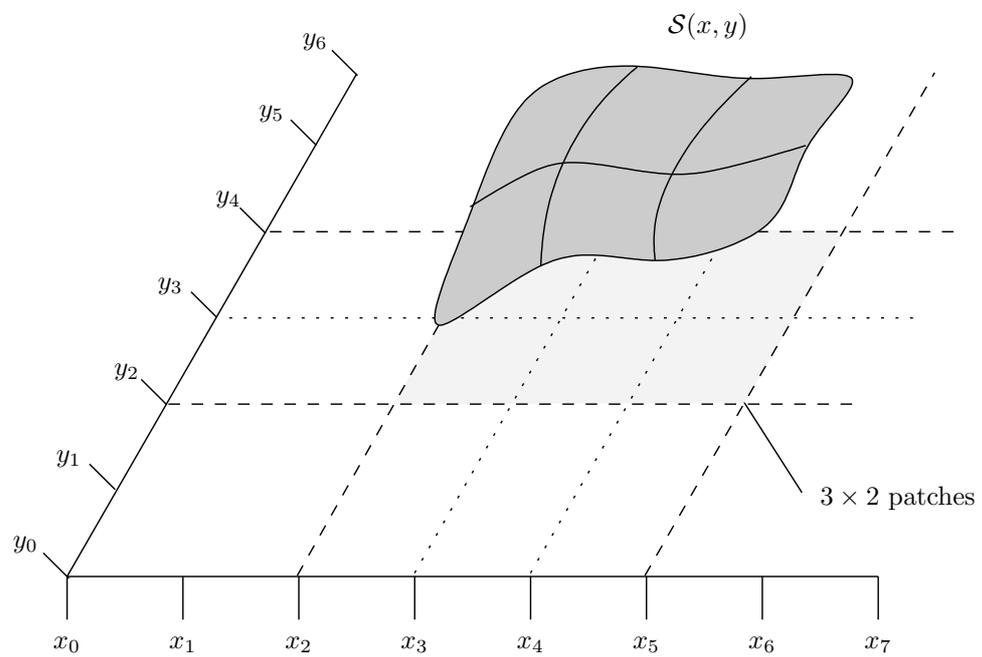


Fig. B.3 — *Example of a uniform quadratic B-spline surface (degree $p = 2$) with $m_i + 1 = 7$ horizontal knots and $m_j + 1 = 6$ vertical knots. The B-spline surface is available on 3×2 patches.*

Appendix C

Rotations Representations and Conversions

A rotation in 3D space can be represented by several ways. This appendix presents the notations, properties and conversions for four of them:

- Rotation matrix: Section C.1.1
- Rotation vector: Section C.1.2
- Euler angles and Cardan-Bryant angles representations (6 different representations) and particularly the one which decompose the rotation along x -axis, then y -axis and at last z -axis: Section C.1.3
- Quaternion: Section C.1.4

It is very useful because it is often wiser to use one specific representation depending on the situation and conversions from one representation to another are commonly necessary. For instance, quaternion representation is often used for the motion recovery (see Section 3.1.3), rotation vector representation is used for bundle adjustment because it is a minimal representation of a rotation, (see Section 2.3.1), *etc.*

C.1 Different Representations

C.1.1 Rotation Matrix

Notation

It is a 3×3 matrix (because in 3D space) and is defined by nine parameters (non minimal representation).

$$\mathbf{R} = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix}$$

Let's note \mathbf{l}_i the i -th row (r_{i1}, r_{i2}, r_{i3}) and \mathbf{c}_i the i -th column (r_{1i}, r_{2i}, r_{3i}) .

General Formulation

A rotation of an angle θ around an axis of unit vector $\mathbf{d} = (d_x \ d_y \ d_z)^\top$ is denoted:

$$\mathbf{R} = \begin{pmatrix} d_x^2(1 - \cos \theta) + \cos \theta & d_x d_y(1 - \cos \theta) - d_z \sin \theta & d_x d_z(1 - \cos \theta) + d_y \sin \theta \\ d_x d_y(1 - \cos \theta) + d_z \sin \theta & d_y^2(1 - \cos \theta) + \cos \theta & d_y d_z(1 - \cos \theta) - d_x \sin \theta \\ d_x d_z(1 - \cos \theta) - d_y \sin \theta & d_y d_z(1 - \cos \theta) + d_x \sin \theta & d_z^2(1 - \cos \theta) + \cos \theta \end{pmatrix} \quad (\text{C.1})^\top$$

In particular, for $\mathbf{d} = (1 \ 0 \ 0)^\top$ (x -axis), $\mathbf{d} = (0 \ 1 \ 0)^\top$ (y -axis) and $\mathbf{d} = (0 \ 0 \ 1)^\top$ (z -axis), we have the following rotation matrices:

$$\begin{aligned} \mathbf{R}_x(\alpha) &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha \\ 0 & \sin \alpha & \cos \alpha \end{pmatrix} \\ \mathbf{R}_y(\beta) &= \begin{pmatrix} \cos \beta & 0 & \sin \beta \\ 0 & 1 & 0 \\ -\sin \beta & 0 & \cos \beta \end{pmatrix} \\ \mathbf{R}_z(\gamma) &= \begin{pmatrix} \cos \gamma & -\sin \gamma & 0 \\ \sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{pmatrix} \end{aligned}$$

with $\mathbf{R}_x(\alpha)$ the rotation of an angle α around x -axis, $\mathbf{R}_y(\beta)$ the rotation of an angle β around y -axis and $\mathbf{R}_z(\gamma)$ the rotation of an angle γ around z -axis. Composition of these three rotations is studied in Section C.1.3.

Properties

A matrix rotation is an orthonormal matrix and thus has its characteristic properties:

$$\begin{aligned} |\mathbf{R}| &= 1 \\ \mathbf{R}^{-1} &= \mathbf{R}^\top \iff \mathbf{R} \cdot \mathbf{R}^\top = \mathbf{I} \\ \forall i, \forall j, \mathbf{l}_i \cdot \mathbf{l}_j &= \mathbf{c}_i \cdot \mathbf{c}_j = \delta_{ij} \text{ with } \begin{cases} \delta_{ij} = 0 & \text{if } i \neq j \\ \delta_{ij} = 1 & \text{if } i = j \end{cases} \end{aligned}$$

Because it represents a rotation, it also has the following properties:

$$\text{trace}(\mathbf{R}) = 1 + 2 \cos \theta \text{ with } \theta \text{ the angle of rotation (see Equ. (C.1))}$$

Composition of two successive rotations consists simply in multiplying the two rotation matrices in the right order. For instance, the matrix rotation of a rotation $R1$ followed by a rotation $R2$ is $\mathbf{R}_{R2}\mathbf{R}_{R1}$.

C.1.2 Rotation Vector

Notation

It is a column vector and is defined by 3 parameters. It is one of the minimal representations of a rotation.

$$\mathbf{n} = \begin{pmatrix} n_x \\ n_y \\ n_z \end{pmatrix}$$

Properties

- The rotation vector represents the rotation axis and its norm represents the angle. Then, we have: $\mathbf{n} = \theta \cdot \mathbf{d}$. For a given rotation matrix \mathbf{R} , recover the angle and axis of rotation is equivalent to compute the associated rotation vector \mathbf{n} (see Section C.2.4).
- The rotation vector is invariant in the rotation since it represents the rotation axis: $\mathbf{R} \cdot \mathbf{n} = \mathbf{n}$. The rotation vector is therefore an eigenvector associated to the eigenvalue $\lambda = 1$ of the rotation matrix and is unique due to its constraint on its norm.
- No particular relationship exist to compose two successive rotations represented by a rotation vector and they have to be converted into another representation first. For computation efficiency, it is better to convert them into quaternions.
- Let's note \mathbf{p}' the rotation of angle θ of the point \mathbf{p} around the axis defined by \mathbf{n} . We have¹:

$$\mathbf{p}' = \mathbf{p} \cdot \cos \theta + \begin{pmatrix} n_x^2 & n_x n_y & n_x n_z \\ n_x n_y & n_y^2 & n_y n_z \\ n_x n_z & n_y n_z & n_z^2 \end{pmatrix} \mathbf{p} (1 - \cos \theta) + \begin{pmatrix} 0 & -n_z & n_y \\ n_z & 0 & -n_x \\ -n_y & n_x & 0 \end{pmatrix} \mathbf{p} \sin \theta$$

¹It is equivalent to compute the result of the rotation using this formula or to convert the rotation vector into a rotation matrix (Section C.2.1) and multiply it by the original point \mathbf{p}

C.1.3 Cardan-Bryant Angles Representations

Notation

Cardan-Bryant representation is defined by 3 parameters (minimal representation) corresponding to the angles of rotation around the different axis: x -axis, y -axis and z -axis. Six possible representations exist, depending on the order of the axis: xyz , xzy , yxz , yzx , zxy and zyx . Euler angles representation is the extension of the Cardan-Bryant representations and allows 6 additional representations: xyx , xzx , yxy , $yzzy$, $zxxz$, $zyzy$.

The Cardan-Bryant angles representation is denoted:

$$\mathbf{cb} = [\alpha \quad \beta \quad \gamma]$$

with α , β and γ the angles around respectively the first, the second and the third axis. From now, the only Cardan-Bryant representation that we will use is the one where the first axis is x -axis, the second is y -axis and the third is z -axis.

Note that even if the Cardan-Bryant representation is minimal, we prefer not using it because for a given rotation, the representation is not unique (for instance, $[\pi \quad 0 \quad 0] = [0 \quad \pi \quad \pi]$). This may cause problems, especially for the conversion from the rotation matrix to Cardan-Bryant angles (see Section C.2.5).

C.1.4 Quaternion

Notation

A quaternion is defined by 4 parameters and is therefore a non-minimal representation of a rotation in 3D space. The parameters are a combination of the 3 coordinates of the axis of rotation and of the angle.

It is commonly denoted:

$$\mathbf{q} = [w \quad x \quad y \quad z] \quad \text{or} \quad \mathbf{q} = [s \quad \mathbf{v}] \quad \text{with } s = w \text{ and } \mathbf{v} = (x \quad y \quad z)^\top$$

Because the quaternion is an extension to the complex numbers, it is sometimes denoted:

$$\mathbf{q} = w + x i + y j + z k \quad \text{with} \quad \begin{cases} i^2 = j^2 = k^2 = -1 \\ ij = -ji = k \\ jk = -kj = i \\ ki = -ik = j \end{cases}$$

Definitions and properties

Let's note $\mathbf{q} = [w \ x \ y \ z] = [s \ \mathbf{v}] = w + xi + yj + zk$ and $\mathbf{q}' = [w' \ x' \ y' \ z'] = [s' \ \mathbf{v}'] = w' + x'i + y'j + z'k$

Conjugate of \mathbf{q} : $\bar{\mathbf{q}} = [w \ -x \ -y \ -z] = [s \ -\mathbf{v}] = w - xi - yj - zk$

$$\text{Multiplication : } \left\{ \begin{array}{l} \mathbf{q}\mathbf{q}' = [ww' - xx' - yy' - zz' \quad wx' + xw' + yz' - zy' \\ \quad wy' + yw' + zx' - xz' \quad wz' + zw' + xy' - yx'] \\ = [ss' - \mathbf{v}\mathbf{v}', s\mathbf{v}' + s'\mathbf{v} + \mathbf{v} \times \mathbf{v}'] \\ \text{where } \times \text{ represents the cross product} \\ = ww' - xx' - yy' - zz' + (wx' + xw' + yz' - zy')i + \\ (wy' + yw' + zx' - xz')j + (wz' + zw' + xy' - yx')k \end{array} \right.$$

Non-commutativity : $\mathbf{q}\mathbf{q}' \neq \mathbf{q}'\mathbf{q}$

Associativity : $(\mathbf{q}\mathbf{q}')\mathbf{q}'' = \mathbf{q}(\mathbf{q}'\mathbf{q}'')$

Norm of \mathbf{q} : $\|\mathbf{q}\| = \sqrt{\mathbf{q}\bar{\mathbf{q}}} = \sqrt{w^2 + x^2 + y^2 + z^2}$

Inversion : $\mathbf{q}^{-1} = \frac{\bar{\mathbf{q}}}{\|\mathbf{q}\|}$

Unit quaternion : \mathbf{q} is a unit quaternion if $\|\mathbf{q}\| = 1$

Normalization of \mathbf{q} : $\mathbf{q}_{\|\cdot\|} = \left[\frac{w}{\|\mathbf{q}\|} \quad \frac{x}{\|\mathbf{q}\|} \quad \frac{y}{\|\mathbf{q}\|} \quad \frac{z}{\|\mathbf{q}\|} \right]$

Composition of rotations : The quaternion of a rotation represented by \mathbf{q} followed by a rotation represented by \mathbf{q}' is $\mathbf{q}'\mathbf{q}$

Rotation of a point \mathbf{p} : $\mathbf{p}' = \mathbf{q}_{\|\cdot\|} \cdot [0, \mathbf{p}] \cdot \mathbf{q}^{-1}$

$$\text{Axis and angle of rotation : } \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \|\mathbf{q}\| \sin \frac{\theta}{2} \cdot \mathbf{d} \quad \text{and} \quad w = \|\mathbf{q}\| \cos \frac{\theta}{2} \tag{C.2}$$

with \mathbf{d} the unit vector of the axis and θ the angle of rotation

C.2 Conversions

C.2.1 Rotation vector \mapsto Rotation matrix

Conversion

$$\mathbf{n} = \begin{pmatrix} n_x \\ n_y \\ n_z \end{pmatrix} \mapsto \begin{cases} \text{If } \mathbf{n} \neq \vec{0}, \\ \text{then } \mathbf{R} = \begin{pmatrix} \frac{n_x^2(1-c_N)}{N^2} + c_N & \frac{n_x n_y(1-c_N)}{N^2} - \frac{n_z s_N}{N} & \frac{n_x n_z(1-c_N)}{N^2} + \frac{n_y s_N}{N} \\ \frac{n_x n_y(1-c_N)}{N^2} + \frac{n_z s_N}{N} & \frac{n_y^2(1-c_N)}{N^2} + c_N & \frac{n_y n_z(1-c_N)}{N^2} - \frac{n_x s_N}{N} \\ \frac{n_x n_z(1-c_N)}{N^2} - \frac{n_y s_N}{N} & \frac{n_y n_z(1-c_N)}{N^2} + \frac{n_x s_N}{N} & \frac{n_z^2(1-c_N)}{N^2} + c_N \end{pmatrix} \\ \text{with } N = \|\mathbf{n}\|, c_N = \cos N \text{ and } s_N = \sin N \\ \text{else } \mathbf{R} = \mathbf{I} \end{cases} \quad (\text{C.3})$$

Demonstration

Section C.1.2 shows that the rotation vector represents the axis of rotation and then, it is collinear to \mathbf{d} . Moreover, its norm is equal to the angle of rotation θ and thus we have:

$$\mathbf{n} = \theta \cdot \mathbf{d} \Rightarrow \begin{cases} \text{If } \theta \neq 0, \\ \text{then } \frac{1}{\theta} \cdot \mathbf{n} = \mathbf{d} \\ \text{else } \mathbf{n} = \vec{0} \end{cases}$$

If $\mathbf{n} \neq \vec{0}$, the matrix rotation can be expressed as a function of n_x , n_y , n_z and $\|\mathbf{n}\|$ from Equ. (C.1):

$$\begin{aligned} \mathbf{R} &= \begin{pmatrix} d_x^2(1 - \cos \theta) + \cos \theta & d_x d_y(1 - \cos \theta) - d_z \sin \theta & d_x d_z(1 - \cos \theta) + d_y \sin \theta \\ d_x d_y(1 - \cos \theta) + d_z \sin \theta & d_y^2(1 - \cos \theta) + \cos \theta & d_y d_z(1 - \cos \theta) - d_x \sin \theta \\ d_x d_z(1 - \cos \theta) - d_y \sin \theta & d_y d_z(1 - \cos \theta) + d_x \sin \theta & d_z^2(1 - \cos \theta) + \cos \theta \end{pmatrix} \\ &= \begin{pmatrix} \frac{n_x^2(1 - \cos \theta)}{\theta^2} + \cos \theta & \frac{n_x n_y(1 - \cos \theta)}{\theta^2} - \frac{n_z \sin \theta}{\theta} & \frac{n_x n_z(1 - \cos \theta)}{\theta^2} + \frac{n_y \sin \theta}{\theta} \\ \frac{n_x n_y(1 - \cos \theta)}{\theta^2} + \frac{n_z \sin \theta}{\theta} & \frac{n_y^2(1 - \cos \theta)}{\theta^2} + \cos \theta & \frac{n_y n_z(1 - \cos \theta)}{\theta^2} - \frac{n_x \sin \theta}{\theta} \\ \frac{n_x n_z(1 - \cos \theta)}{\theta^2} - \frac{n_y \sin \theta}{\theta} & \frac{n_y n_z(1 - \cos \theta)}{\theta^2} + \frac{n_x \sin \theta}{\theta} & \frac{n_z^2(1 - \cos \theta)}{\theta^2} + \cos \theta \end{pmatrix} \\ &= \begin{pmatrix} \frac{n_x^2(1 - \cos \|\mathbf{n}\|)}{\|\mathbf{n}\|^2} + \cos \|\mathbf{n}\| & \frac{n_x n_y(1 - \cos \|\mathbf{n}\|)}{\|\mathbf{n}\|^2} - \frac{n_z \sin \|\mathbf{n}\|}{\|\mathbf{n}\|} & \frac{n_x n_z(1 - \cos \|\mathbf{n}\|)}{\|\mathbf{n}\|^2} + \frac{n_y \sin \|\mathbf{n}\|}{\|\mathbf{n}\|} \\ \frac{n_x n_y(1 - \cos \|\mathbf{n}\|)}{\|\mathbf{n}\|^2} + \frac{n_z \sin \|\mathbf{n}\|}{\|\mathbf{n}\|} & \frac{n_y^2(1 - \cos \|\mathbf{n}\|)}{\|\mathbf{n}\|^2} + \cos \|\mathbf{n}\| & \frac{n_y n_z(1 - \cos \|\mathbf{n}\|)}{\|\mathbf{n}\|^2} - \frac{n_x \sin \|\mathbf{n}\|}{\|\mathbf{n}\|} \\ \frac{n_x n_z(1 - \cos \|\mathbf{n}\|)}{\|\mathbf{n}\|^2} - \frac{n_y \sin \|\mathbf{n}\|}{\|\mathbf{n}\|} & \frac{n_y n_z(1 - \cos \|\mathbf{n}\|)}{\|\mathbf{n}\|^2} + \frac{n_x \sin \|\mathbf{n}\|}{\|\mathbf{n}\|} & \frac{n_z^2(1 - \cos \|\mathbf{n}\|)}{\|\mathbf{n}\|^2} + \cos \|\mathbf{n}\| \end{pmatrix} \end{aligned}$$

If $\mathbf{n} = \vec{0}$, the rotation is null and the rotation matrix is the identity: $\mathbf{R} = \mathbf{I}$.

C.2.2 Cardan-Bryant \mapsto Rotation matrix

Conversion

$$[\alpha \ \beta \ \gamma] \mapsto \begin{pmatrix} \cos \beta \cos \gamma & -\cos \alpha \sin \gamma + \cos \gamma \sin \alpha \sin \beta & \sin \alpha \sin \gamma + \cos \alpha \cos \gamma \sin \beta \\ \cos \beta \sin \gamma & \cos \alpha \cos \gamma + \sin \alpha \sin \beta \sin \gamma & -\cos \gamma \sin \alpha + \cos \alpha \sin \beta \sin \gamma \\ -\sin \beta & \cos \beta \sin \alpha & \cos \alpha \cos \beta \end{pmatrix} \quad (\text{C.4})$$

Demonstration

Cardan-Bryant representation considers a rotation as the composition of the three following rotations: a rotation of angle α around the x -axis, then a rotation of angle β around y -axis and at last, a rotation of angle γ around z -axis. Then, we have:

$$\begin{aligned} \mathbf{R}_{xyz} &= \mathbf{R}_z \cdot \mathbf{R}_y \cdot \mathbf{R}_x \\ &= \begin{pmatrix} \cos \gamma & -\sin \gamma & 0 \\ \sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos \beta & 0 & \sin \beta \\ 0 & 1 & 0 \\ -\sin \beta & 0 & \cos \beta \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha \\ 0 & \sin \alpha & \cos \alpha \end{pmatrix} \\ &= \begin{pmatrix} \cos \beta \cos \gamma & -\cos \alpha \sin \gamma + \cos \gamma \sin \alpha \sin \beta & \sin \alpha \sin \gamma + \cos \alpha \cos \gamma \sin \beta \\ \cos \beta \sin \gamma & \cos \alpha \cos \gamma + \sin \alpha \sin \beta \sin \gamma & -\cos \gamma \sin \alpha + \cos \alpha \sin \beta \sin \gamma \\ -\sin \beta & \cos \beta \sin \alpha & \cos \alpha \cos \beta \end{pmatrix} \end{aligned}$$

C.2.3 Quaternion \mapsto Rotation matrix

Conversion

$$\mathbf{q} = [w \ x \ y \ z] \mapsto \frac{1}{\|\mathbf{q}\|^2} \begin{pmatrix} w^2 + x^2 - y^2 - z^2 & 2xy - 2wz & 2xz + 2wy \\ 2xy + 2wz & w^2 - x^2 + y^2 - z^2 & 2yz - 2wx \\ 2xz - 2wy & 2yz + 2wx & w^2 - x^2 - y^2 + z^2 \end{pmatrix}$$

In particular, if $\|\mathbf{q}\|^2 = w^2 + x^2 + y^2 + z^2 = 1$, we have:

$$\mathbf{q} = [w \ x \ y \ z] \mapsto \begin{pmatrix} 1 - 2y^2 - 2z^2 & 2xy - 2wz & 2xz + 2wy \\ 2xy + 2wz & 1 - 2x^2 - 2z^2 & 2yz - 2wx \\ 2xz - 2wy & 2yz + 2wx & 1 - 2x^2 - 2y^2 \end{pmatrix} \quad (\text{C.5})$$

Then, for computation efficiency, it is always preferable to normalize the quaternion before converting it.

Demonstration

Section C.1.4 shows that the quaternion parameters can be expressed as a function of \mathbf{d} and θ . On the contrary, if $d_x, d_y, d_z, \cos \theta$ and $\sin \theta$ can be expressed as a function of x, y, z and w , we will be able to replace the former by the latter in Equ. (C.1) and then, obtain the rotation matrix associated to a quaternion.

$$\text{Equ. (C.2)} \Rightarrow \begin{cases} \theta = 2 \arccos \frac{w}{\|\mathbf{q}\|} \\ d_x = \frac{\frac{x}{\|\mathbf{q}\|}}{\sin(\arccos \frac{w}{\|\mathbf{q}\|})} \\ d_y = \frac{\frac{y}{\|\mathbf{q}\|}}{\sin(\arccos \frac{w}{\|\mathbf{q}\|})} \\ d_z = \frac{\frac{z}{\|\mathbf{q}\|}}{\sin(\arccos \frac{w}{\|\mathbf{q}\|})} \end{cases}$$

Let's note $W = \frac{w}{\|\mathbf{q}\|}, X = \frac{x}{\|\mathbf{q}\|}, Y = \frac{y}{\|\mathbf{q}\|}, Z = \frac{z}{\|\mathbf{q}\|}$ and using the following trigonometric identities:

$$\sin(2a) = 2 \sin a \cos a ; \quad \cos(2a) = 2 \cos^2 a - 1 ; \quad \sin(\arccos a) = \sqrt{1 - a^2}$$

we obtain:

$$\begin{cases} \cos \theta = 2W^2 - 1 \\ \sin \theta = 2W\sqrt{1 - W^2} \\ d_x = \frac{X}{\sqrt{1 - W^2}} \\ d_y = \frac{Y}{\sqrt{1 - W^2}} \\ d_z = \frac{Z}{\sqrt{1 - W^2}} \end{cases}$$

These correspondences can now be applied in Equ. (C.1):

$$\begin{aligned} \mathbf{R}_{\mathbf{q}} &= \begin{pmatrix} d_x^2(1 - \cos \theta) + \cos \theta & d_x d_y(1 - \cos \theta) - d_z \sin \theta & d_x d_z(1 - \cos \theta) + d_y \sin \theta \\ d_x d_y(1 - \cos \theta) + d_z \sin \theta & d_y^2(1 - \cos \theta) + \cos \theta & d_y d_z(1 - \cos \theta) - d_x \sin \theta \\ d_x d_z(1 - \cos \theta) - d_y \sin \theta & d_y d_z(1 - \cos \theta) + d_x \sin \theta & d_z^2(1 - \cos \theta) + \cos \theta \end{pmatrix} \\ &= \begin{pmatrix} 2X^2 + 2W^2 - 1 & 2XY - 2WZ & 2XZ + 2WY \\ 2XY + 2WZ & 2Y^2 + 2W^2 - 1 & 2YZ - 2WX \\ 2XZ - 2WY & 2YZ + 2WX & 2Z^2 + 2W^2 - 1 \end{pmatrix} \\ &= \begin{pmatrix} X^2 + W^2 - Y^2 - Z^2 & 2XY - 2WZ & 2XZ + 2WY \\ 2XY + 2WZ & Y^2 + W^2 - X^2 - Z^2 & 2YZ - 2WX \\ 2XZ - 2WY & 2YZ + 2WX & Z^2 + W^2 - X^2 - Y^2 \end{pmatrix} \\ &= \frac{1}{\|\mathbf{q}\|^2} \begin{pmatrix} w^2 + x^2 - y^2 - z^2 & 2xy - 2wz & 2xz + 2wy \\ 2xy + 2wz & w^2 - x^2 + y^2 - z^2 & 2yz - 2wx \\ 2xz - 2wy & 2yz + 2wx & w^2 - x^2 - y^2 + z^2 \end{pmatrix} \end{aligned}$$

C.2.4 Rotation matrix \mapsto Rotation vector

Conversion

$$\mathbf{R} = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} \mapsto \left\{ \begin{array}{l} \text{If } \mathbf{R} = \mathbf{I}, \\ \text{then } \mathbf{n} = \vec{0} \\ \text{else if } r_{11} + r_{22} + r_{33} \neq -1, \\ \\ \text{then } \mathbf{n} = \begin{pmatrix} (r_{32} - r_{23}) \frac{\arccos\left(\frac{r_{11}+r_{22}+r_{33}-1}{2}\right)}{\sqrt{4-(r_{11}+r_{22}+r_{33}-1)^2}} \\ (r_{13} - r_{31}) \frac{\arccos\left(\frac{r_{11}+r_{22}+r_{33}-1}{2}\right)}{\sqrt{4-(r_{11}+r_{22}+r_{33}-1)^2}} \\ (r_{21} - r_{12}) \frac{\arccos\left(\frac{r_{11}+r_{22}+r_{33}-1}{2}\right)}{\sqrt{4-(r_{11}+r_{22}+r_{33}-1)^2}} \end{pmatrix} \\ \\ \text{else } \mathbf{n} = \begin{pmatrix} n_x \\ n_y \\ n_z \end{pmatrix} \text{ with } \left\{ \begin{array}{l} n_x = -\sqrt{\frac{\pi^2}{2}}(r_{11} + 1) \\ \text{If } r_{12} > 0, \\ \text{then } n_y = -\sqrt{\frac{\pi^2}{2}}(r_{22} + 1) \\ \text{else } n_y = \sqrt{\frac{\pi^2}{2}}(r_{22} + 1) \\ \text{If } r_{13} > 0, \\ \text{then } n_z = -\sqrt{\frac{\pi^2}{2}}(r_{33} + 1) \\ \text{else } n_z = \sqrt{\frac{\pi^2}{2}}(r_{33} + 1) \end{array} \right. \end{array} \right.$$

Demonstration

First, the case $\mathbf{R} = \mathbf{I} \Rightarrow \mathbf{n} = \vec{0}$ is obvious and does not require a demonstration.

Section C.2.1 shows that a rotation matrix can be expressed as a function of the parameters of a rotation vector (see Equ. (C.3)). For a rotation matrix not equal to the identity (corresponding to $\mathbf{n} \neq \vec{0}$):

$$\left(\begin{array}{lll} r_{11} = \frac{n_x^2(1-\cos\|\mathbf{n}\|)}{\|\mathbf{n}\|^2} + \cos\|\mathbf{n}\| & r_{12} = \frac{n_x n_y(1-\cos\|\mathbf{n}\|)}{\|\mathbf{n}\|^2} - \frac{n_z \sin\|\mathbf{n}\|}{\|\mathbf{n}\|} & r_{13} = \frac{n_x n_z(1-\cos\|\mathbf{n}\|)}{\|\mathbf{n}\|^2} + \frac{n_y \sin\|\mathbf{n}\|}{\|\mathbf{n}\|} \\ r_{21} = \frac{n_x n_y(1-\cos\|\mathbf{n}\|)}{\|\mathbf{n}\|^2} + \frac{n_z \sin\|\mathbf{n}\|}{\|\mathbf{n}\|} & r_{22} = \frac{n_y^2(1-\cos\|\mathbf{n}\|)}{\|\mathbf{n}\|^2} + \cos\|\mathbf{n}\| & r_{23} = \frac{n_y n_z(1-\cos\|\mathbf{n}\|)}{\|\mathbf{n}\|^2} - \frac{n_x \sin\|\mathbf{n}\|}{\|\mathbf{n}\|} \\ r_{31} = \frac{n_x n_z(1-\cos\|\mathbf{n}\|)}{\|\mathbf{n}\|^2} - \frac{n_y \sin\|\mathbf{n}\|}{\|\mathbf{n}\|} & r_{32} = \frac{n_y n_z(1-\cos\|\mathbf{n}\|)}{\|\mathbf{n}\|^2} + \frac{n_x \sin\|\mathbf{n}\|}{\|\mathbf{n}\|} & r_{33} = \frac{n_z^2(1-\cos\|\mathbf{n}\|)}{\|\mathbf{n}\|^2} + \cos\|\mathbf{n}\| \end{array} \right)$$

leading to the following relationships:

$$\left\{ \begin{array}{l} r_{32} - r_{23} = 2n_x \frac{\sin\|\mathbf{n}\|}{\|\mathbf{n}\|} \\ r_{13} - r_{31} = 2n_y \frac{\sin\|\mathbf{n}\|}{\|\mathbf{n}\|} \\ r_{21} - r_{12} = 2n_z \frac{\sin\|\mathbf{n}\|}{\|\mathbf{n}\|} \end{array} \right.$$

The parameters of the rotation vector are then known up to a coefficient $2 \frac{\sin\|\mathbf{n}\|}{\|\mathbf{n}\|}$. The more direct method to obtain this coefficient is as follows:

$$r_{11} + r_{22} + r_{33} - 1 = 2 \cos \|\mathbf{n}\| \Rightarrow 2 \frac{\sin \|\mathbf{n}\|}{\|\mathbf{n}\|} = \frac{\sqrt{4 - (r_{11} + r_{22} + r_{33} - 1)^2}}{\arccos\left(\frac{r_{11} + r_{22} + r_{33} - 1}{2}\right)}$$

Before dividing by this coefficient, we need to check if it is not null:

$$\frac{\sqrt{4 - (r_{11} + r_{22} + r_{33} - 1)^2}}{\arccos\left(\frac{r_{11} + r_{22} + r_{33} - 1}{2}\right)} \neq 0 \iff \begin{cases} r_{11} + r_{22} + r_{33} \neq 3 \\ r_{11} + r_{22} + r_{33} \neq -1 \end{cases}$$

The statement $r_{11} + r_{22} + r_{33} = 3$ is not possible here. Indeed, the trace of a matrix being equal to $1 + 2 \cos \theta$, this would correspond to $\theta = 0 \pmod{2\pi}$, *i.e.* a null rotation, and this is not possible since the matrix rotation is not the identity.

The statement $r_{11} + r_{22} + r_{33} = -1$ corresponds to $\|\mathbf{n}\| = \theta = \pi \pmod{2\pi}$. The rotation matrix becomes:

$$\begin{pmatrix} r_{11} = \frac{2n_x^2}{\pi^2} - 1 & r_{12} = \frac{2n_x n_y}{\pi^2} & r_{13} = \frac{2n_x n_z}{\pi^2} \\ r_{21} = \frac{2n_x n_y}{\pi^2} & r_{22} = \frac{2n_y^2}{\pi^2} & r_{23} = \frac{2n_y n_z}{\pi^2} \\ r_{31} = \frac{2n_x n_z}{\pi^2} & r_{32} = \frac{2n_y n_z}{\pi^2} & r_{33} = \frac{2n_z^2}{\pi^2} \end{pmatrix}$$

and using r_{11}, r_{22} and r_{33} , the parameters n_x, n_y and n_z of the rotation vector \mathbf{n} can be identified but with an ambiguity of their signs. Because the angle of rotation is $\|\mathbf{n}\| = \theta = \pi \pmod{2\pi}$, it is equivalent to have the direction of the axis of rotation $\mathbf{n} = \begin{pmatrix} n_x & n_y & n_z \end{pmatrix}^\top$ or $\mathbf{n} = \begin{pmatrix} -n_x & -n_y & -n_z \end{pmatrix}^\top$. Therefore, in order to determine the missing signs, it is possible to fix one sign (n_x for instance) and deduce the signs of the other parameters (n_y and n_z) using the signs of r_{12}, r_{13} or r_{23} .

Finally, we have several possibilities:

$$\begin{array}{l} \text{If } r_{11} + r_{22} + r_{33} \neq -1 \\ \text{then } \begin{cases} n_x = (r_{32} - r_{23}) \frac{\arccos\left(\frac{r_{11} + r_{22} + r_{33} - 1}{2}\right)}{\sqrt{4 - (r_{11} + r_{22} + r_{33} - 1)^2}} \\ n_y = (r_{13} - r_{31}) \frac{\arccos\left(\frac{r_{11} + r_{22} + r_{33} - 1}{2}\right)}{\sqrt{4 - (r_{11} + r_{22} + r_{33} - 1)^2}} \\ n_z = (r_{21} - r_{12}) \frac{\arccos\left(\frac{r_{11} + r_{22} + r_{33} - 1}{2}\right)}{\sqrt{4 - (r_{11} + r_{22} + r_{33} - 1)^2}} \end{cases} \\ \text{else } \begin{cases} n_x = -\sqrt{\frac{\pi^2}{2}(r_{11} + 1)} \text{ (} n_x \text{ is fixed negative)} \\ \text{If } r_{12} > 0, \text{ then } n_y = -\sqrt{\frac{\pi^2}{2}(r_{22} + 1)} \text{ else } n_y = \sqrt{\frac{\pi^2}{2}(r_{22} + 1)} \\ \text{If } r_{13} > 0, \text{ then } n_z = -\sqrt{\frac{\pi^2}{2}(r_{33} + 1)} \text{ else } n_z = \sqrt{\frac{\pi^2}{2}(r_{33} + 1)} \end{cases} \end{array}$$

C.2.5 Rotation matrix \mapsto Cardan-Bryant

Conversion

$$\mathbf{R} = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} \mapsto \begin{cases} \text{If } r_{11} = 0, r_{21} = 0, r_{32} = 0 \text{ and } r_{33} = 0, \\ \text{then if } r_{31} = 1 \\ \quad \text{then } \begin{cases} \alpha = \arctan\left(\frac{r_{12}}{r_{13}}\right) = \arctan\left(\frac{r_{12}}{r_{22}}\right) \\ \beta = \frac{\pi}{2} \\ \gamma = 0 \end{cases} \\ \quad \text{else } r_{31} = -1 \text{ and } \begin{cases} \alpha = \arctan\left(\frac{r_{12}}{r_{13}}\right) = \arctan\left(\frac{-r_{12}}{r_{22}}\right) \\ \beta = -\frac{\pi}{2} \\ \gamma = 0 \end{cases} \\ \text{else } \begin{cases} \alpha = \arctan\left(\frac{r_{32}}{r_{33}}\right) \\ \beta = \arctan\left(\frac{-r_{31}}{\sqrt{r_{11}^2 + r_{21}^2}}\right) \\ \gamma = \arctan\left(\frac{r_{21}}{r_{11}}\right) \end{cases} \end{cases}$$

Demonstration

Section C.2.2 shows that a matrix rotation can be expressed with only the parameters of the Cardan-Bryant representation. Then, several methods exist to recover the Cardan-Bryant angles α , β and γ from Equ. (C.4) but the one presented here has the advantage of avoiding sign ambiguities by taking relationships giving the tangent value of the Cardan-Bryant angles.

Equ. (C.4):

$$\begin{pmatrix} r_{11} = \cos \beta \cos \gamma & r_{12} = -\cos \alpha \sin \gamma + \cos \gamma \sin \alpha \sin \beta & r_{13} = \sin \alpha \sin \gamma + \cos \alpha \cos \gamma \sin \beta \\ r_{21} = \cos \beta \sin \gamma & r_{22} = \cos \alpha \cos \gamma + \sin \alpha \sin \beta \sin \gamma & r_{23} = -\cos \gamma \sin \alpha + \cos \alpha \sin \beta \sin \gamma \\ r_{31} = -\sin \beta & r_{32} = \cos \beta \sin \alpha & r_{33} = \cos \alpha \cos \beta \end{pmatrix}$$

leads to:

$$\begin{cases} \frac{r_{21}}{r_{11}} = \tan \gamma \text{ if } \cos \beta \neq 0 \\ \frac{r_{32}}{r_{33}} = \tan \alpha \text{ if } \cos \beta \neq 0 \\ \sqrt{r_{11}^2 + r_{21}^2} = \cos \beta \text{ and } r_{31} = -\sin \beta \Rightarrow \tan \beta = \frac{-r_{31}}{\sqrt{r_{11}^2 + r_{21}^2}} \text{ if } \cos \beta \neq 0 \end{cases}$$

In order to identify the parameters α , β and γ , the specific case where $\cos \beta = 0$ has to be treated (this problem is known as ‘‘Gimbal lock’’). Actually, it is not possible to determine a unique solution for the angles α and γ in this case and one is expressed as a function of the other. We choose to fix $\gamma = 0$ and determine the value of α using r_{12} and r_{13} or r_{12} and r_{22} .

$$\cos \beta = 0 \Leftrightarrow r_{11} = 0, r_{21} = 0, r_{32} = 0 \text{ and } r_{33} = 0$$

At last, we have:

If $r_{11} = 0, r_{21} = 0, r_{32} = 0$ et $r_{33} = 0$,

then if $r_{31} = 1$,

then $\begin{cases} \beta = \frac{\pi}{2} \\ \text{we fix: } \gamma = 0 \\ \text{and we have: } \alpha = \arctan\left(\frac{r_{12}}{r_{13}}\right) = \arctan\left(\frac{r_{12}}{r_{22}}\right) \end{cases}$

else $r_{31} = -1$ and $\begin{cases} \beta = -\frac{\pi}{2} \\ \text{we fix: } \gamma = 0 \\ \text{and we have: } \alpha = \arctan\left(\frac{r_{12}}{r_{13}}\right) = \arctan\left(\frac{-r_{12}}{r_{22}}\right) \end{cases}$

C.2.6 Rotation matrix \mapsto Quaternion

Conversion

$$\begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} \mapsto \begin{cases} \text{If } r_{11} + r_{22} + r_{33} + 1 \neq 0, \text{ then} \\ \mathbf{q} = \begin{bmatrix} \frac{\sqrt{r_{11}+r_{22}+r_{33}+1}}{2} & \frac{r_{32}-r_{23}}{2\sqrt{r_{11}+r_{22}+r_{33}+1}} & \frac{r_{13}-r_{31}}{2\sqrt{r_{11}+r_{22}+r_{33}+1}} & \frac{r_{21}-r_{12}}{2\sqrt{r_{11}+r_{22}+r_{33}+1}} \end{bmatrix} \\ \text{else if } r_{11} - r_{22} - r_{33} + 1 \neq 0, \text{ then} \\ \mathbf{q} = \begin{bmatrix} \frac{r_{32}-r_{23}}{2\sqrt{r_{11}-r_{22}-r_{33}+1}} & \frac{\sqrt{r_{11}-r_{22}-r_{33}+1}}{2} & \frac{r_{21}+r_{12}}{2\sqrt{r_{11}-r_{22}-r_{33}+1}} & \frac{r_{13}+r_{31}}{2\sqrt{r_{11}-r_{22}-r_{33}+1}} \end{bmatrix} \\ \text{else if } -r_{11} + r_{22} - r_{33} + 1 \neq 0, \text{ then} \\ \mathbf{q} = \begin{bmatrix} \frac{r_{13}-r_{31}}{2\sqrt{-r_{11}+r_{22}-r_{33}+1}} & \frac{r_{21}+r_{12}}{2\sqrt{-r_{11}+r_{22}-r_{33}+1}} & \frac{\sqrt{-r_{11}+r_{22}-r_{33}+1}}{2} & \frac{r_{23}+r_{32}}{2\sqrt{-r_{11}+r_{22}-r_{33}+1}} \end{bmatrix} \\ \text{else we know that } -r_{11} - r_{22} + r_{33} + 1 \neq 0, \text{ and} \\ \mathbf{q} = \begin{bmatrix} \frac{r_{21}-r_{12}}{2\sqrt{-r_{11}-r_{22}+r_{33}+1}} & \frac{r_{13}+r_{31}}{2\sqrt{-r_{11}-r_{22}+r_{33}+1}} & \frac{r_{23}+r_{32}}{2\sqrt{-r_{11}-r_{22}+r_{33}+1}} & \frac{\sqrt{-r_{11}-r_{22}+r_{33}+1}}{2} \end{bmatrix} \end{cases}$$

Actually, as we know that $r_{11} + r_{22} + r_{33} + 1$, $r_{11} - r_{22} - r_{33} + 1$, $-r_{11} + r_{22} - r_{33} + 1$ and $-r_{11} - r_{22} + r_{33} + 1$ cannot be simultaneously all null, we prefer using the following algorithm:

$$\begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} \mapsto \begin{cases} 1. \text{ Calculate the maximum of } r_{11} + r_{22} + r_{33} + 1, r_{11} - r_{22} - r_{33} + 1, \\ -r_{11} + r_{22} - r_{33} + 1 \text{ and } -r_{11} - r_{22} + r_{33} + 1 \\ 2. \text{ If the maximum is } r_{11} + r_{22} + r_{33} + 1, \text{ then} \\ \mathbf{q} = \begin{bmatrix} \frac{\sqrt{r_{11}+r_{22}+r_{33}+1}}{2} & \frac{r_{32}-r_{23}}{2\sqrt{r_{11}+r_{22}+r_{33}+1}} & \frac{r_{13}-r_{31}}{2\sqrt{r_{11}+r_{22}+r_{33}+1}} & \frac{r_{21}-r_{12}}{2\sqrt{r_{11}+r_{22}+r_{33}+1}} \end{bmatrix} \\ \text{else if the maximum is } r_{11} - r_{22} - r_{33} + 1, \text{ then} \\ \mathbf{q} = \begin{bmatrix} \frac{r_{32}-r_{23}}{2\sqrt{r_{11}-r_{22}-r_{33}+1}} & \frac{\sqrt{r_{11}-r_{22}-r_{33}+1}}{2} & \frac{r_{21}+r_{12}}{2\sqrt{r_{11}-r_{22}-r_{33}+1}} & \frac{r_{13}+r_{31}}{2\sqrt{r_{11}-r_{22}-r_{33}+1}} \end{bmatrix} \\ \text{else if the maximum is } -r_{11} + r_{22} - r_{33} + 1, \text{ then} \\ \mathbf{q} = \begin{bmatrix} \frac{r_{13}-r_{31}}{2\sqrt{-r_{11}+r_{22}-r_{33}+1}} & \frac{r_{21}+r_{12}}{2\sqrt{-r_{11}+r_{22}-r_{33}+1}} & \frac{\sqrt{-r_{11}+r_{22}-r_{33}+1}}{2} & \frac{r_{23}+r_{32}}{2\sqrt{-r_{11}+r_{22}-r_{33}+1}} \end{bmatrix} \\ \text{else the maximum is } -r_{11} - r_{22} + r_{33} + 1, \text{ and} \\ \mathbf{q} = \begin{bmatrix} \frac{r_{21}-r_{12}}{2\sqrt{-r_{11}-r_{22}+r_{33}+1}} & \frac{r_{13}+r_{31}}{2\sqrt{-r_{11}-r_{22}+r_{33}+1}} & \frac{r_{23}+r_{32}}{2\sqrt{-r_{11}-r_{22}+r_{33}+1}} & \frac{\sqrt{-r_{11}-r_{22}+r_{33}+1}}{2} \end{bmatrix} \end{cases}$$

Demonstration

Section C.2.3 shows that a rotation matrix can be expressed with the parameters of a unit quaternion (see Equ. (C.5)). Then, it should be possible to extract from any rotation matrix the parameters of an equivalent unit quaternion.

$$\text{Equ. (C.5) : } \begin{pmatrix} r_{11} = 1 - 2y^2 - 2z^2 & r_{12} = 2xy - 2wz & r_{13} = 2xz + 2wy \\ r_{21} = 2xy + 2wz & r_{22} = 1 - 2x^2 - 2z^2 & r_{23} = 2yz - 2wx \\ r_{31} = 2xz - 2wy & r_{32} = 2yz + 2wx & r_{33} = 1 - 2x^2 - 2y^2 \end{pmatrix}$$

where $w^2 + x^2 + y^2 + z^2 = 1$.

This leads to:

$$\begin{cases} r_{11} + r_{22} + r_{33} + 1 = 4(1 - x^2 - y^2 - z^2) = 4w^2 \\ r_{11} - r_{22} - r_{33} + 1 = 4x^2 \\ -r_{11} + r_{22} - r_{33} + 1 = 4y^2 \\ -r_{11} - r_{22} + r_{33} + 1 = 4z^2 \end{cases}$$

A first approach would consist in using these 4 equalities to obtain w, x, y and z with an ambiguity of sign for each parameter and finding other relationships in the matrix to remove the ambiguities. However, we prefer using another method to avoid the sign ambiguities:

$$\begin{cases} r_{11} + r_{22} + r_{33} + 1 = 4w^2 \\ r_{32} - r_{23} = 4wx \\ r_{13} - r_{31} = 4wy \\ r_{21} - r_{12} = 4wz \end{cases} \Rightarrow \begin{cases} w = \pm \frac{\sqrt{r_{11} + r_{22} + r_{33} + 1}}{2} \\ \text{if } w \neq 0, \text{ then } x = \frac{r_{32} - r_{23}}{4w} \\ \text{if } w \neq 0, \text{ then } y = \frac{r_{13} - r_{31}}{4w} \\ \text{if } w \neq 0, \text{ then } z = \frac{r_{21} - r_{12}}{4w} \end{cases}$$

The advantage here is that the sign of w does not need to be known because if its sign is wrong, the signs of x, y and z will be wrong as well and we know that the quaternions $\mathbf{q} = [x \ y \ z \ w]$ and $\mathbf{q} = [-x \ -y \ -z \ -w]$ are equivalent.

However, this is only possible if $w \neq 0 \Leftrightarrow r_{11} + r_{22} + r_{33} + 1 \neq 0$. Three other equivalent methods exist, each one being available under a particular condition:

$$\begin{cases} r_{11} - r_{22} - r_{33} + 1 = 4x^2 \\ r_{32} - r_{23} = 4wx \\ r_{21} + r_{12} = 4xy \\ r_{13} + r_{31} = 4xz \end{cases} \Rightarrow \begin{cases} x = \pm \frac{\sqrt{r_{11} - r_{22} - r_{33} + 1}}{2} \\ \text{if } x \neq 0, \text{ then } w = \frac{r_{32} - r_{23}}{4x} \\ \text{if } x \neq 0, \text{ then } y = \frac{r_{21} + r_{12}}{4x} \\ \text{if } x \neq 0, \text{ then } z = \frac{r_{13} + r_{31}}{4x} \end{cases}$$

$$\begin{cases} -r_{11} + r_{22} - r_{33} + 1 = 4y^2 \\ r_{13} - r_{31} = 4wy \\ r_{21} + r_{12} = 4xy \\ r_{23} + r_{32} = 4yz \end{cases} \Rightarrow \begin{cases} y = \pm \frac{\sqrt{-r_{11} + r_{22} - r_{33} + 1}}{2} \\ \text{if } y \neq 0, \text{ then } w = \frac{r_{13} - r_{31}}{4y} \\ \text{if } y \neq 0, \text{ then } x = \frac{r_{21} + r_{12}}{4y} \\ \text{if } y \neq 0, \text{ then } z = \frac{r_{23} + r_{32}}{4y} \end{cases}$$

$$\begin{cases} -r_{11} - r_{22} + r_{33} + 1 = 4z^2 \\ r_{21} - r_{12} = 4wz \\ r_{13} + r_{31} = 4xz \\ r_{23} + r_{32} = 4yz \end{cases} \Rightarrow \begin{cases} z = \pm \frac{\sqrt{-r_{11}-r_{22}+r_{33}+1}}{2} \\ \text{if } z \neq 0, \text{ then } w = \frac{r_{21}-r_{12}}{4z} \\ \text{if } z \neq 0, \text{ then } x = \frac{r_{13}+r_{31}}{4z} \\ \text{if } z \neq 0, \text{ then } y = \frac{r_{23}+r_{32}}{4z} \end{cases}$$

Note that there is always at least one method available. Indeed, the parameters cannot be simultaneously all null because the following equation system:

$$\begin{cases} r_{11} + r_{22} + r_{33} + 1 = 0 \\ r_{11} - r_{22} - r_{33} + 1 = 0 \\ -r_{11} + r_{22} - r_{33} + 1 = 0 \\ -r_{11} - r_{22} + r_{33} + 1 = 0 \end{cases}$$

does not have any solution.

The general approach consists then in finding a non-null parameter and deducing the other parameters with the appropriate method:

$$\left\{ \begin{array}{l} \text{If } r_{11} + r_{22} + r_{33} + 1 \neq 0, \\ \text{then } \mathbf{q} = \left[\frac{\sqrt{r_{11}+r_{22}+r_{33}+1}}{2} \quad \frac{r_{32}-r_{23}}{2\sqrt{r_{11}+r_{22}+r_{33}+1}} \quad \frac{r_{13}-r_{31}}{2\sqrt{r_{11}+r_{22}+r_{33}+1}} \quad \frac{r_{21}-r_{12}}{2\sqrt{r_{11}+r_{22}+r_{33}+1}} \right] \\ \text{else if } r_{11} - r_{22} - r_{33} + 1 \neq 0, \\ \text{then } \mathbf{q} = \left[\frac{r_{32}-r_{23}}{2\sqrt{r_{11}-r_{22}-r_{33}+1}} \quad \frac{\sqrt{r_{11}-r_{22}-r_{33}+1}}{2} \quad \frac{r_{21}+r_{12}}{2\sqrt{r_{11}-r_{22}-r_{33}+1}} \quad \frac{r_{13}+r_{31}}{2\sqrt{r_{11}-r_{22}-r_{33}+1}} \right] \\ \text{else if } -r_{11} + r_{22} - r_{33} + 1 \neq 0, \\ \text{then } \mathbf{q} = \left[\frac{r_{13}-r_{31}}{2\sqrt{-r_{11}+r_{22}-r_{33}+1}} \quad \frac{r_{21}+r_{12}}{2\sqrt{-r_{11}+r_{22}-r_{33}+1}} \quad \frac{\sqrt{-r_{11}+r_{22}-r_{33}+1}}{2} \quad \frac{r_{23}+r_{32}}{2\sqrt{-r_{11}+r_{22}-r_{33}+1}} \right] \\ \text{else we know that } -r_{11} - r_{22} + r_{33} + 1 \neq 0, \\ \text{and } \mathbf{q} = \left[\frac{r_{21}-r_{12}}{2\sqrt{-r_{11}-r_{22}+r_{33}+1}} \quad \frac{r_{13}+r_{31}}{2\sqrt{-r_{11}-r_{22}+r_{33}+1}} \quad \frac{r_{23}+r_{32}}{2\sqrt{-r_{11}-r_{22}+r_{33}+1}} \quad \frac{\sqrt{-r_{11}-r_{22}+r_{33}+1}}{2} \right] \end{array} \right.$$

C.2.7 Rotation vector \mapsto Quaternion

Conversion

$$\mathbf{n} = \begin{pmatrix} n_x \\ n_y \\ n_z \end{pmatrix} \Rightarrow \begin{cases} \text{If } \|\mathbf{n}\| = 0 \\ \text{then } \mathbf{q} = [1 \ 0 \ 0 \ 0] \\ \text{else } \mathbf{q} = \left[\cos \frac{\|\mathbf{n}\|}{2} \quad \frac{n_x}{\|\mathbf{n}\|} \sin \frac{\|\mathbf{n}\|}{2} \quad \frac{n_y}{\|\mathbf{n}\|} \sin \frac{\|\mathbf{n}\|}{2} \quad \frac{n_z}{\|\mathbf{n}\|} \sin \frac{\|\mathbf{n}\|}{2} \right] \end{cases}$$

Demonstration

The demonstration is here very simple. For a unit quaternion:

$$\text{Equ. (C.2)} \Rightarrow \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \sin \frac{\theta}{2} \cdot \mathbf{d} \text{ et } w = \cos \frac{\theta}{2}$$

Moreover, $\|\mathbf{n}\| = \theta$ and $\mathbf{d} = \frac{\mathbf{n}}{\|\mathbf{n}\|}$ if $\|\mathbf{n}\| \neq 0$. Then:

$$\begin{cases} \text{If } \|\mathbf{n}\| \neq 0 \\ \text{then } \mathbf{q} = \begin{bmatrix} \cos \frac{\|\mathbf{n}\|}{2} & \frac{n_x}{\|\mathbf{n}\|} \sin \frac{\|\mathbf{n}\|}{2} & \frac{n_y}{\|\mathbf{n}\|} \sin \frac{\|\mathbf{n}\|}{2} & \frac{n_z}{\|\mathbf{n}\|} \sin \frac{\|\mathbf{n}\|}{2} \end{bmatrix} \\ \text{else the rotation is null and } \mathbf{q} = [1 \ 0 \ 0 \ 0] \end{cases}$$

C.2.8 Quaternion \mapsto Rotation vector

Conversion

$$\mathbf{q} = [w \ x \ y \ z] \Rightarrow \begin{cases} \text{If } x = 0, y = 0 \text{ and } z = 0 \\ \text{then } \mathbf{n} = \vec{0} \\ \text{elsen } \mathbf{n} = \frac{\arccos(2\frac{w^2}{\|\mathbf{q}\|^2} - 1)}{\sqrt{\|\mathbf{q}\|^2 - w^2}} \begin{pmatrix} x \\ y \\ z \end{pmatrix} \end{cases}$$

In particular, if \mathbf{q} is a unit quaternion and $w \neq 1$:

$$\mathbf{q} = [w \ x \ y \ z] \Rightarrow \mathbf{n} = \frac{\arccos(2w^2 - 1)}{\sqrt{1 - w^2}} \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

Demonstration

Here again, the demonstration is very simple. Using $\mathbf{n} = \theta\mathbf{d}$ and the different relationships of Section C.2.3:

$$\text{Si } \|\mathbf{q}\| \neq w^2, \begin{cases} d_x = \frac{x}{\sqrt{\|\mathbf{q}\|^2 - w^2}} \\ d_y = \frac{y}{\sqrt{\|\mathbf{q}\|^2 - w^2}} \\ d_z = \frac{z}{\sqrt{\|\mathbf{q}\|^2 - w^2}} \end{cases}$$

else $x = 0, y = 0$ and $z = 0 \Rightarrow \mathbf{n} = \vec{0}$ because the rotation is null

Moreover:

$$\cos \theta = 2\frac{w^2}{\|\mathbf{q}\|^2} - 1 \Rightarrow \theta = \arccos\left(2\frac{w^2}{\|\mathbf{q}\|^2} - 1\right)$$

leading at last to:

$$\begin{cases} \text{If } x = 0, y = 0 \text{ and } z = 0 \\ \text{then } \mathbf{n} = \vec{0} \\ \text{else } \mathbf{n} = \frac{\arccos(2\frac{w^2}{\|\mathbf{q}\|^2} - 1)}{\sqrt{\|\mathbf{q}\|^2 - w^2}} \begin{pmatrix} x \\ y \\ z \end{pmatrix} \end{cases}$$

Appendix D

SEM Calibration - Experimental setup

D.1 Preparation

D.1.1 SEM Settings

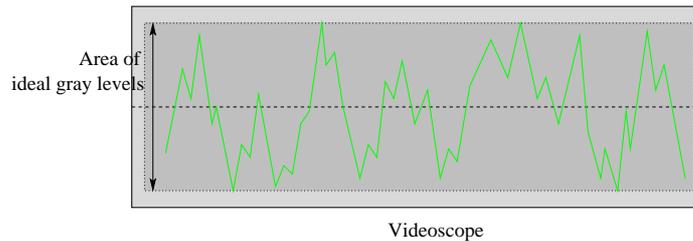
- Set the digital contrast and digital gamma to 1 and the digital brightness to 0.
- Save images in an uncompressed 8-bit format (BMP, TIFF, ...). Image resolution must be at least 1024x884.
- Set the data-bar to:

HV	Spot	WD	Det	Mag	Scan	Date Time	┌ micron-bar ┐
----	------	----	-----	-----	------	--------------	----------------

- Note the dwell time value

D.1.2 Operating Conditions

- Working Distance ideally set to 10 mm
- Lowest accelerating voltage possible to obtain a sharp and detailed image
- Scan speed greater than 30 ms per line
- Spot size: usually 3
- BSE detector if possible (if the pattern with the current specimen is good enough for correlation)
- Setup the contrast/brightness in order to have the maximum gray levels used without saturating (check the videoscope)



ONCE SET UP, DO NOT CHANGE ANY PARAMETERS DURING THE EXPERIMENT!!!

D.2 Experimental Procedure

General recommendations:

- after each motion of the stage (translation or rotation) and before acquiring the image, wait at least 30 seconds so that the SEM stage is stabilized.
- do the entire experiment as fast as possible so that the operating conditions are almost the same from the first to the last acquired image.
- if the image goes out of focus during the tilt sequence, do not change the working distance! Refocus by moving the stage in Z direction.
- choose the initial position so that there is a specific mark in the center of the image. That will be very helpful for repositioning of the specimen during the rotation sequence.

The experiment procedure consists of the following steps (this is obviously an example and rotation and tilt angle values depend generally on the experiment):

1. Acquire an image in initial position and name it calib_000a. Acquire consecutively a second image calib_000b.
2. Rotate the stage by 90° and acquire 2 new consecutive images calib_001a and calib_001b.
3. Rotate again by 90° (total angle: 180°) and acquire 2 new consecutive images calib_002a and calib_002b.
4. Rotate a last time by 90° (total angle: 270°) and acquire images calib_003a and calib_003b.
5. Tilt the stage by an angle as big as possible without shrinking the original image by more than a few percent. The value depends obviously on the magnification (generally

between 3° for high magnification and 20° for low magnification). Center the image approximately at the position it was before the tilt (this is why a specific mark on the specimen in the center of the image at initial position can be helpful). Repeat the steps 1. to 4., acquire 8 images and name them calib_004a, ... , calib_007b

6. Tilt the stage approximately by the opposite angle. Recenter the image, and repeat again steps 1. to 4. to acquire 8 new images and name them calib_008a, ... , calib_011b

Image Filename	Rotation	Tilt
calib_000a.tif	0°	0°
calib_000b.tif	0°	0°
calib_001a.tif	90°	0°
calib_001b.tif	90°	0°
calib_002a.tif	180°	0°
calib_002b.tif	180°	0°
calib_003a.tif	270°	0°
calib_003b.tif	270°	0°

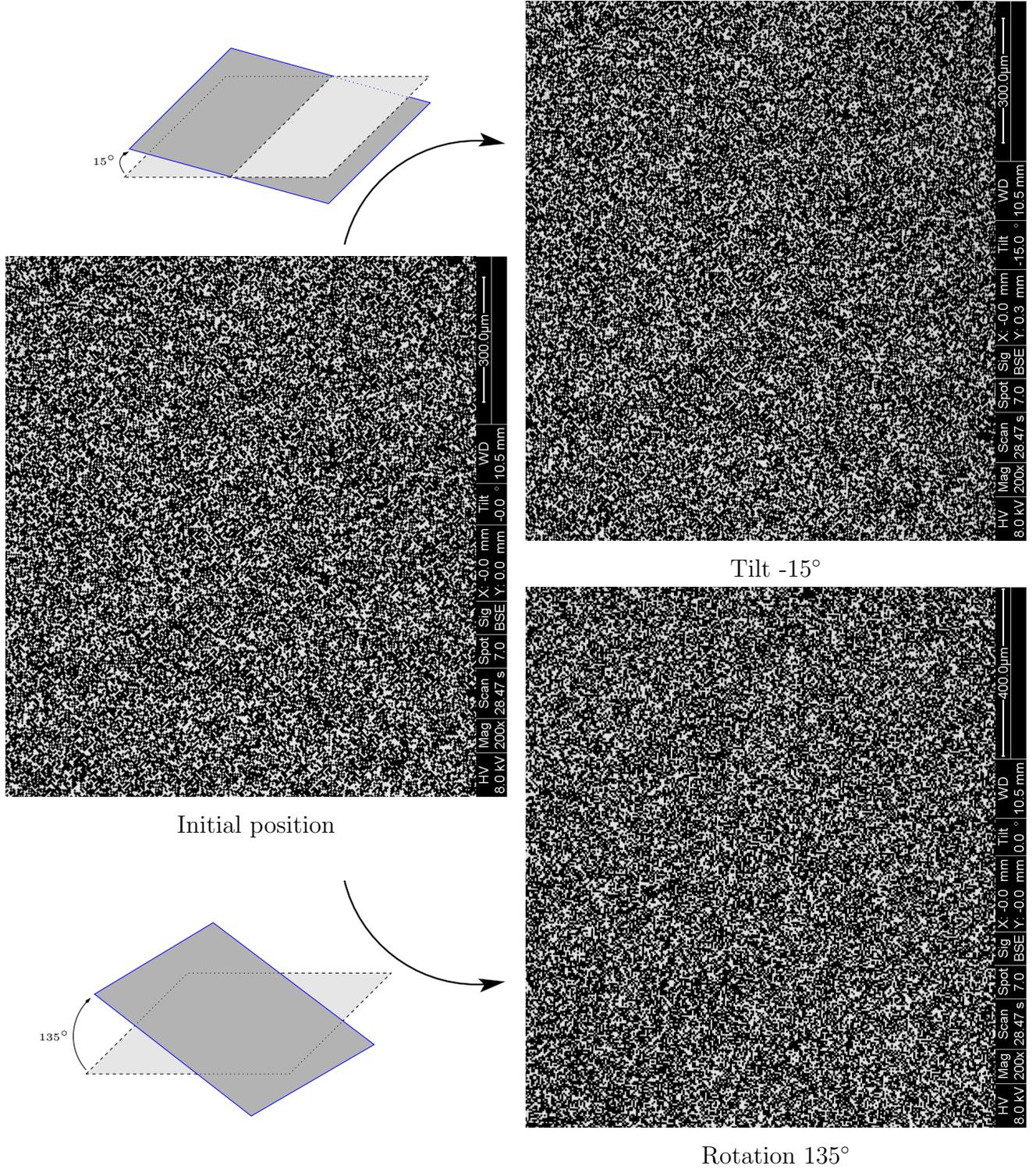
Image Filename	Rotation	Tilt
calib_004a.tif	0°	10°
calib_004b.tif	0°	10°
calib_005a.tif	90°	10°
calib_005b.tif	90°	10°
calib_006a.tif	180°	10°
calib_006b.tif	180°	10°
calib_007a.tif	270°	10°
calib_007b.tif	270°	10°

Image Filename	Rotation	Tilt
calib_008a.tif	0°	-10°
calib_008b.tif	0°	-10°
calib_009a.tif	90°	-10°
calib_009b.tif	90°	-10°
calib_010a.tif	180°	-10°
calib_010b.tif	180°	-10°
calib_011a.tif	270°	-10°
calib_011b.tif	270°	-10°

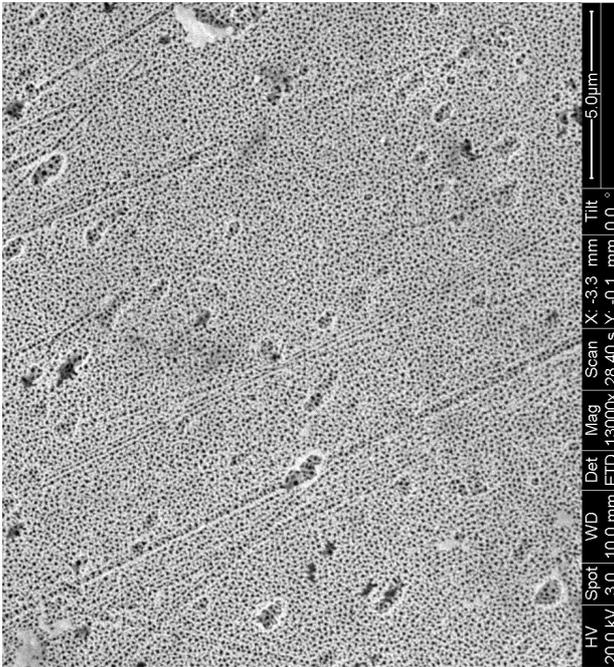
Example of image sequence

D.3 Example of images

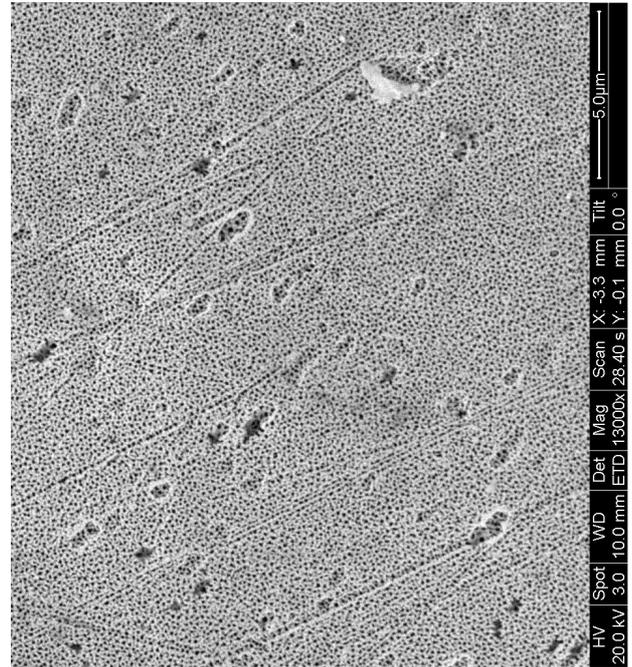
Speckle pattern of gold deposited on an aluminum wafer by micro-lithography



Gold coated anodized aluminum

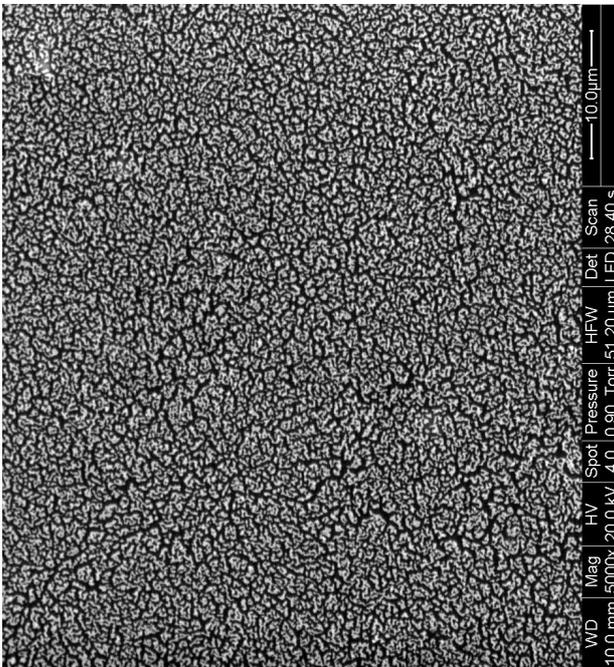


Translation X: 3 μ m

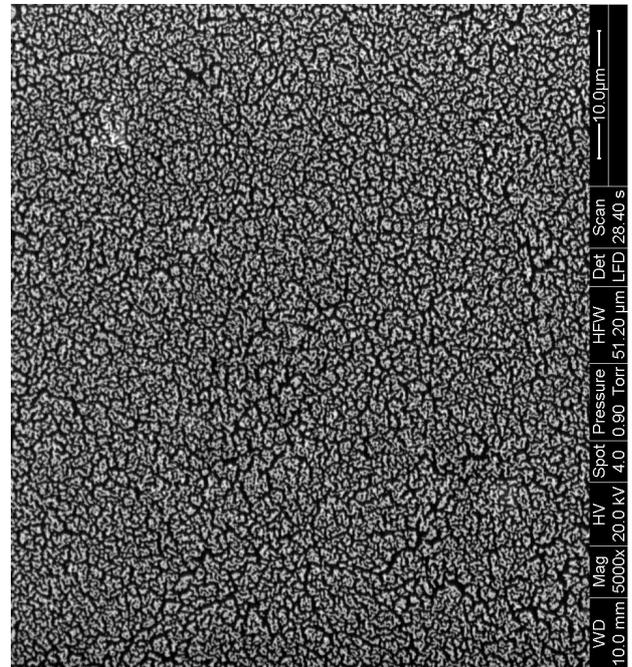


Translation X: 5 μm, Y: 5 μm

Epoxy substrate



Translation X: 5 μ m



Translation X: 5 μm, Y: 5 μm

General Bibliography

- [ABBC94] Lucien Allais, Michel Bornert, Thierry Bretheau, and Daniel Caldemaison. Experimental characterization of the local strain field in a heterogeneous elastoplastic material. *Acta Metallurgica et Materialia*, 42(11):3865–3880, 1994.
- [AGO00] B. Aubé, D. Garcia, and J.-J. Orteu. Un scanner 3D pour la modélisation d’objets, simple à utiliser et n’utilisant qu’une seule caméra – An easy-to-use 3D scanner for object modelling using a single camera. In *SCANNING 2000: The Scanning European Meeting Point, Fifth Edition*, Paris (France), May 2000.
- [AHD⁺00] M. Agrawal, D. Harwood, R. Duraiswami, L. Davis, and P. Luther. Three Dimensional Ultrastructure from Transmission Electron Microscope Tilt Series. In *Second Indian Conference on Vision, Graphics and Image Processing (ICVGIP 2000)*, Bangalore (India), 2000.
- [AN88] J. Aggarwal and N. Nandhakumar. On the computation of motion from sequences of images – a review. In *Proceedings of IEEE*, volume 76, pages 917–935, 1988.
- [AS03] X. Armangue and J. Salvi. Overall View Regarding Fundamental Matrix Estimation. *Image and Vision Computing*, 21(2):205–220, 2003.
- [Bar92] C. Barlier. Le procédé de prototypage rapide par STRATOCONCEPTION[®]. In *Proceedings of the first "Assises Européennes du Prototypage Rapide"*, Palaiseau (France), Jun 1992.
- [BB94] R. Brand, P. Mohr and P. Bobet. Distorsions optiques: corrections dans un modèle projectif. In *Proceedings of 9th Congrès AFCET, Reconnaissances des Formes et Intelligence Artificielle (RFIA'94)*, pages 87–98, Paris (France), Jan 1994.

- [BDM04] M. Bornert, P. Doumalin, and E. Maire. Mesure tridimensionnelle des champs de déformation par corrélation d'images microtomographiques. In *Photomécanique 2004*, pages 33–40, Albi (France), 2004.
- [Ben98] Raouf Benjemaa. *Traitement de données 3D denses acquises sur des objets réels complexes*. PhD thesis, ENST, Paris, Oct 1998.
- [Bey92] Horst A. Beyer. Accurate Calibration of CCD-cameras. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR'92)*, pages 96–101, Urbana Champaign, (IL, USA), Jun 1992.
- [BFGM95] C. Barlier, U. Feltes, D. Gasser, and F. Muller. STRATOCONCEPTION[®], Rapid Prototyping for die-forging tooling. In *Proceedings of 28th International Symposium on Automotive Technology and Automation (ISATA)*, Stuttgart (Germany), Sep 1995.
- [Bro58] D.C. Brown. A solution to the general problem of multiple station analytical stereotriangulation. Technical Report 23, RCA Data Reduction, Patrick Air Force base, Florida (USA), 1958.
- [Bro71] D. C. Brown. Lens Distortion for Close-Range Photogrammetry. *Photometric Engineering*, 37(8):855–866, 1971.
- [BS99] R. Benjemaa and F. Schmitt. Fast global registration of 3D sampled surfaces using a multi-z-buffer technique. *Image and Vision Computing*, 17(2):113–123, 1999.
- [BSFS99] B.K. Bay, T.S. Smith, D.P. Fyhrie, and M. Saad. Digital volume correlation: three-dimensional strain mapping using X-ray tomography. *Experimental Mechanics*, 39(3):218–227, 1999.
- [Can86] John F. Canny. A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–714, 1986.
- [CGS⁺03] Nicolas Cornille, Dorian Garcia, Michael A. Sutton, Stephen R. McNeill, and Jean-José Orteu. Automated 3D Reconstruction Using a Scanning Electron Microscope. In *SEM Conference on Experimental and Applied Mechanics*, Charlotte (NC, USA), Jun 2003.
- [CGS⁺04a] Nicolas Cornille, Dorian Garcia, Michael A. Sutton, Stephen R. McNeill, and Jean-José Orteu. Calibrage d'un MEB en vue de la mesure précise de formes et de déformations 3D. In *Proceedings of Colloque Photomécanique 2004*, Albi (France), 2004.

-
- [CGS⁺04b] Nicolas Cornille, Dorian Garcia, Michael A. Sutton, Stephen R. McNeill, and Jean-José Orteu. Initial Experimental Results Using an ESEM for Metrology. In *SEM Conference on Experimental and Applied Mechanics*, Costa Mesa (CA, USA), Jun 2004.
- [CLVWdV02] E. Cecchi, J.M. Lavest, and B. Van Wyk de Vries. Videogrammetric reconstruction applied to volcanology: perspectives for a new measurement technique in volcano monitoring. *Image Analysis and Stereology*, 21(1):31–36, 2002.
- [Cor01] Nicolas Cornille. Numérisation 3D d’objets observés au MEB (Microscope Electronique à Balayage). Master’s thesis, Institut National des Sciences Appliquées de Toulouse, Jun 2001.
- [Cot94] J.C. Cottier. Extraction et appariements robustes des points d’intérêts de deux images non étalonnées. Technical report, LIFIA-IMAG-INRIA, 1994.
- [CRSP85] T. C. Chu, W. F. Ranson, M. A. Sutton, and W. H. Peters. Application of Digital Image Correlation Techniques to Experimental Mechanics. *Experimental Mechanics*, 25(3):232–244, 1985.
- [CSG⁺04] Nicolas Cornille, Hubert W. Schreier, Dorian Garcia, Michael A. Sutton, Stephen R. McNeill, and Jean-José Orteu. Gridless Calibration of Distorted Visual Sensors. In *Proceedings of Colloque Photomécanique 2004*, Albi (France), 2004.
- [CSI] Vic2D[®] and Vic3D[®], Correlated Solutions Inc., <http://www.correlatedsolutions.com>.
- [CSM⁺04] S. A. Collette, M. A. Sutton, P. Miney, A. P. Reynolds, Xiaodong Li, P. E. Colavita, W. A. Scrivens, Y. Luo, T. Sudarshan, P. Muzykov, and M. L. Myrick. Development of patterns for nanoscale strain measurements: I. fabrication of imprinted au webs for polymeric materials. *Nanotechnology*, 15(12):1812–1817, December 2004.
- [DF95] F. Devernay and O. Faugeras. From Projective to Euclidean Reconstruction. Technical Report RR-2725, INRIA, Nov 1995.
- [Dou00] Pascal Doumalin. *Microextensométrie locale par corrélation d’images numériques*. PhD thesis, Ecole Polytechnique, 2000.
- [Far93] G. Farin. *Curves and Surfaces for Computer Aided Geometric Design – A Practical Guide*. Academic Press, 1993.
-

- [Fau93] Olivier Faugeras. *Three-Dimensional Computer Vision: A Geometric Viewpoint*. MIT Press, 1993. ISBN 0-262-06158-9.
- [Fau95] Olivier Faugeras. Stratification of 3D Vision: Projective, Affine and Metric Representations. *Journal of the Optical Society of America A*, 12(3):465–484, Mar 1995.
- [FLP01] O. Faugeras, Q.-T. Luong, and T. Papadopoulou. *The Geometry of Multiple Images: The Laws That Govern The Formation of Images of A Scene and Some of Their Applications*. MIT Press, 2001.
- [FM95] O. D. Faugeras and B. Mourrain. On the geometry and algebra of the point and line correspondences between n images. In *Proceedings of the 5th International Conference on Computer Vision (ICCV'95)*, pages 951–962, Boston (MA, USA), 1995.
- [For94] Wolfgang Forstner. A framework for low-level feature extraction. In *Proceedings of the 3rd European Conference on Computer Vision (ECCV'94)*, pages 383–394, Stockholm (Sweden), 1994.
- [Gar01] Dorian Garcia. *Mesure de formes et de champs de déplacements tridimensionnels par stéréo-corrélation d'images*. PhD thesis, Institut National Polytechnique de Toulouse (France), Dec 2001.
- [GH99] K. Galanulis and A. Hofmann. Determination of Forming Limit Diagrams Using an Optical Measurement System. In *7th International Conference on Sheet Metal*, pages 245–252, Erlangen (Germany), Sep 1999.
- [GOP02] Dorian Garcia, Jean-José Orteu, and Luc Penazzi. A combined temporal tracking and stereo-correlation technique for accurate measurement of 3D displacements: application to sheet metal forming. *Journal of Materials Processing Technology*, 125-126:736–742, Jan 2002.
- [Gru97] Armin Gruen. Fundamentals of videogrammetry: a review. *Human Movement Science*, 16(2-3):155–187, 1997.
- [GSV97] N. Gracias and J. Santos-Victor. Robust estimation of the fundamental matrix and stereo correspondences. In *International Symposium on Intelligent Robotic Systems*, Stockholm (Sweden), Jul 1997.
- [HA00] M. Hemmleb and J. Albertz. Microtopography - The Photogrammetric Determination of Friction Surfaces. In *IAPRS, Vol. XXXIII*, Amsterdam, 2000.

-
- [Har94] Richard I. Hartley. Lines and Points in Three Views - An Integrated Approach. In *ARPA Image Understanding Workshop*, 1994.
- [Har95] Richard Hartley. In Defence of the 8-point Algorithm. In *5th International Conference on Computer Vision (ICCV'95)*, pages 1064–1070, Boston (MA, USA), Jun 1995.
- [Har98] Richard Hartley. Computation of the Quadrifocal Tensor. In *5th European Conference on Computer Vision (ECCV'98)*, volume 1406, pages 20–35, Freiburg (Germany), Jun 1998.
- [HAS⁺96] M. Hemmleb, J. Albertz, M. Schubert, A. Gleichmann, and J. M. Köhler. Digital microphotogrammetry with the scanning electron microscope. In *International Society for Photogrammetry and Remote Sensing*, 1996.
- [HMS96] J. D. Helm, S. R. McNeill, and M. A. Sutton. Improved 3D Image Correlation for Surface Displacement Measurement. *Optical Engineering*, 35(7):1911–1920, 1996.
- [HN94] T. Huang and A. Netravali. Motion and structure from feature correspondences: a review. In *Proceedings of IEEE*, volume 82, pages 252–268, 1994.
- [HRVDH⁺92] F. Heitger, L. Rosenthaler, R. Von Der Heydt, E. Peterhans, and O. Kuebler. Simulation of neural contour mechanism: from simple to end-stopped cells. *Vision Research*, 32(5):963–981, 1992.
- [HS88] C. Harris and M. Stephens. A Combined Corner and Edge Detector. In *Alvey Vision Conference*, pages 147–151, 1988.
- [HS97] M. Hemmleb and M. Schubert. Digital microphotogrammetry - Determination of the topography of microstructures by scanning electron microscope. In *Second Turkish-German Joint Geodetic Days*, pages 745–752, Berlin (Germany), May 1997.
- [HSV90] R. Horaud, T. Skordas, and F. Veillon. Finding geometric and relational structures in an image. In *Proceedings of the 1st European Conference on Computer Vision (ECCV'90)*, pages 374–384, Antibes (France), 1990.
- [HZ00] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.
- [JJ95] Drahomira Janova and Jiri Jan. Robust Surface Reconstruction from Stereo SEM Images. In V. Hlavac and R. Sara, editors, *Computer Analysis of*
-

- Images and Patterns (CAIP'95)*, pages 900–905, Berlin (Germany), 1995. Springer.
- [JL01] I.K. Jung and S. Lacroix. A robust interest points matching algorithm. In *8th International Conference on Computer Vision (ICCV'2001)*, Vancouver (Canada), Jul 2001.
- [KJC90] Z. L. Khan-Jetter and T. C. Chu. Three-Dimensional Displacement Measurements Using Digital Image Correlation and Photogrammic Analysis. *Experimental Mechanics*, 30(1):10–16, 1990.
- [Kul61] H. E. Kulsrud. A practical technique for the determination of the optimum relaxation factor of the successive over-relaxation method. In *Communications of the ACM*, volume 4, pages 184–187, 1961.
- [Lav96] Stéphane Laveau. *Géométrie d'un système de n caméras. Théorie, estimation et applications*. PhD thesis, Ecole Polytechnique, May 1996.
- [LBVP⁺04] Guy Le Besnerais, Fabien Vignon, Jean-Louis Pouchou, Denis Boivin, and Pierre Beauchene. Surface Reconstruction from Multiple SEM Images. In *Workshop on Applications of Computer Vision 2004*, Orlando (FL, USA), Dec 2004.
- [LCS94] P. F. Luo, Y. J. Chao, and M. A. Sutton. Application of Stereo Vision to 3D Deformation Analysis in Fracture Mechanics. *Optical Engineering*, 33:981, 1994.
- [LCSP93] P. F. Luo, Y. J. Chao, M. A. Sutton, and W. H. Peters. Accurate Measurement of Three-Dimensional Deformable and Rigid Bodies Using Computer Vision. *Experimental Mechanics*, 33(2):123–132, 1993.
- [LF96] Q.-T. Luong and O. Faugeras. The fundamental matrix: Theory, algorithms, and stability analysis. *International Journal of Computer Vision*, 17(1):43–76, 1996.
- [LH81] Hugh Christopher Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135, Sep 1981.
- [LSP03] H.S. Lee, G.H. Shin, and H.D. Park. Digital Surface Modeling for Assessment of Weathering Rate of Weathered Rock in Stone Monuments. In *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume 34, Ancona (Italy), Jul 2003.

-
- [LT87] P. Lascaux and R. Théodor. *Analyse numérique matricielle appliquée à l'art de l'ingénieur*. Masson, 1987. ISBN 2-90360-760-5.
- [LTCY98] A.J. Lacey, S. Thacker, S. Crossley, and R.B. Yates. A multi-stage approach to the dense estimation of disparity from stereo SEM images. *Image and Vision Computing*, 16:373–383, 1998.
- [LTY96] A. J. Lacey, N. A. Thacker, and R. B. Yates. Surface Approximation from Industrial SEM Images. In *British Machine Vision Conference (BMVC'96)*, pages 725–734, 1996.
- [LVD98] J-M. Lavest, M. Viala, and M. Dhome. Do we really need an accurate calibration pattern to achieve a reliable camera calibration? In *Proceedings of 5th European Conference on Computer Vision - ECCV'98*, volume 1, pages 158–174, Freiburg (Germany), Jun 1998.
- [MDD96] E. Mazza, G. Danuser, and J. Dual. Light Optical Measurements in Microbars with Nanometer Resolution. *Microsystem Technologies*, 2:83–91, 1996.
- [MeX] MeX[®] software, Alicona Imaging, <http://www.alicon.com/>.
- [MKWJ99] H. L. Mitchell, H. T. Kniest, and O. Won-Jin. Digital Photogrammetry and Microscope Photographs. *Photogrammetric Record*, 16(94):695–704, 1999.
- [MNSW⁺83] S. R. Mc Neill, M. A. Sutton, W. J. Wolters, W. H. Peters, and W. F. Ranson. Determination of Displacements Using an Improved Digital Correlation Method. *Image and Vision Computing*, 1(3):1333–1339, 1983.
- [MZ92] J.-L. Mundy and A. Zisserman. *Geometric Invariance in Computer Vision*. MIT Press, 1992.
- [Nie98] H. B. Nielsen. Cubic splines. Lecture notes, Department of Mathematical Modeling, Technical University of Denmark, 1998.
- [OGD97] J.-J. Orteu, V. Garric, and M. Devy. Camera Calibration for 3-D Reconstruction: Application to the Measure of 3-D Deformations on Sheet Metal Parts. In *European Symposium on Lasers, Optics and Vision in Manufacturing*, Munich (Germany), Jun 1997.
- [Ost53] A. Ostrowski. On Over and Under Relaxation in the Theory of the Cyclic Single Step Iteration. *Mathematical Tables and Other Aids to Computation*, 7(43):152–159, 1953.
-

- [Pen55] R.A. Penrose. A Generalized Inverse for Matrices. In *Proceedings of the Cambridge Philosophical Society*, volume 51, pages 406–413, 1955.
- [Peu93] Bernard Peuchot. Camera Virtual Equivalent Model – 0.01 Pixel Detector. In *Proceedings of Computerized Medical Imaging and Graphics*, volume 17, pages 289–294, Jul 1993.
- [Peu94] B. Peuchot. Utilisation de détecteurs subpixels dans la modélisation d’une caméra. In *Proceedings of 9th Congrès AFCET, Reconnaissances des Formes et Intelligence Artificielle (RFIA ’94)*, pages 691–695, Paris (France), Jan 1994.
- [Pol99] M. Pollefeys. *Self-Calibration and Metric 3D Reconstruction from Uncalibrated Image Sequences*. PhD thesis, Katholieke Universiteit Leuven, May 1999.
- [Pol00] Marc Pollefeys. Tutorial on 3D Modeling from Images. In *European Conference on Computer Vision (ECCV’2000)*, Dublin (Ireland), Jun 2000.
- [PR82] W. H. Peters and W. F. Ranson. Digital Imaging Techniques in Experimental Stress Analysis. *Optical Engineering*, 21(3):427–432, 1982.
- [RA94] V. Rodin and A. Ayache. Axial stereovision: modelization and comparison between two calibration methods. In *IEEE International Conference on Image Processing*, Austin (TX, USA), Nov 1994.
- [RAS93] O. Ravn, N. A. Andersen, and A. T. Sorensen. Auto-calibration in Automation Systems using Vision. In *3rd International Symposium on Experimental Robotics (ISER’93)*, pages 206–218, Kyoto (Japan), Oct 1993.
- [RL87] P. Rousseeuw and A. Leriy. *Robust Regression and Outlier Detection*. John Wiley & Sons, 1987.
- [RS05] A. Restrepo-Specht. *Modélisation d’objets 3D par construction incrémentale d’un maillage triangulaire, dans un contexte robotique*. PhD thesis, Université Paul Sabatier, Jan 2005.
- [RSDS02] A. Restrepo-Specht, M. Devy, and A. Sappa. Robustness to initial conditions in registration of range images using the ICP algorithm. In *Proceedings of 3rd International Symposium on Robotics and Automation, ISRA’2002*, Toluca (Mexico), Sep 2002.

-
- [RWT00] R.G. Richards, M. Wieland, and M. Textor. Advantages of stereo imaging of metallic surfaces with low voltage backscattered electrons in a field emission scanning electron microscope. *Journal of Microscopy*, 199:115–123, Aug 2000.
- [SA91] M. E. Spetakis and J. Aloimonos. A multi-frame approach to visual motion perception. *International Journal of Computer Vision*, 16(3):245–255, 1991.
- [SAM] SAMx[®], 3D_TOPx package, <http://www.samx.com>.
- [Sas94] A. Sashua. Trilinearity in visual recognition by alignment. In *Proceedings of the 3rd European Conference on Computer Vision (ECCV'94)*, volume 1, pages 479–484, Stockholm (Sweden), May 1994.
- [SB97] S. M. Smith and J. M. Brady. SUSAN - a new approach to low-level image processing. *International Journal of Computer Vision*, 23(1):45–78, 1997.
- [Sch03] Hubert Schreier. *Investigation of Two and Three-Dimensional Image Correlation Techniques with Applications in Experimental Mechanics*. PhD thesis, University of South Carolina, 2003.
- [SCP⁺86] M.A. Sutton, M. Cheng, W. H. Peters, Y. J. Chao, and S. R. McNeill. Application of an Optimized Digital Correlation Method to Planar Deformation Analysis. *Image and Vision Computing*, 4(3):143–153, 1986.
- [SCTB90] M. A. Sutton, T. L. Chae, J. L. Turner, and H. A. Bruck. Development of a Computer Vision Methodology for the Analysis of Surface Deformations in Magnified Images. In George F. Van Der Voort, editor, *MiCon 90: Advances in Video Technology for Microstructural Control, ASTM STP 1094*, pages 109–132, Philadelphia (PA, USA), 1990. American Society for Testing and Materials.
- [SDL96] P. Sayd, M. Dhome, and J.-M. Lavest. Axial Stereovision for recovering straight cylinders. In *Proceedings of IEEE Conference on Robotics and Cybernetics CESA'96*, pages 560–564, Lille (France), Jul 1996.
- [SGS04] H. Schreier, D. Garcia, and M. A. Sutton. Advances in Light Microscope Stereo Vision. *Experimental Mechanics*, 44(3):278–288, 2004.
- [SHL95] W. Shih, Y-P. Hung, and W-S. Lin. When should we consider lens distortion in camera calibration? *Pattern Recognition*, 28(3):447–461, 1995.
- [SMB00] C. Schmid, R. Mohr, and C. Bauckhage. Evaluation of Interest Point Detectors. *International Journal of Computer Vision*, 37(2):151–172, 2000.
-

- [SMHS00] M. A. Sutton, S. R. McNeill, J. D. Helm, and H. W. Schreier. Computer Vision Applied to Shape and Deformation Measurement. In *International Conference on Trends in Optical Nondestructive Testing and Inspection*, pages 571–589, Lugano (Switzerland), 2000.
- [SRK⁺02] O. Sinram, M. Ritter, S. Kleindiek, A. Schertel, H. Hohenberg, and J. Albertz. Calibration of an SEM, Using a Nano Positioning Tilting Table and a Microscopic Calibration Pyramid. In *ISPRS Commission V Symposium*, pages 210–215, Corfu (Greece), 2002.
- [SS99] P. Synnergren and M. Sjö Dahl. A Stereoscopic Digital Speckle Photography System for 3D Displacement Field Measurements. *Optics and Lasers in Engineering*, 31:425–443, 1999.
- [SWP⁺83] M. A. Sutton, W. J. Wolters, W. H. Peters, W. F. Ranson, and S. R. McNeill. Determination of Displacements Using an Improved Digital Correlation Method. *Image and Vision Computing*, 21:133–139, 1983.
- [TH84] R. Y. Tsai and T. S. Huang. Uniqueness and estimation of three-dimensional motion parameters of rigid objects with curved surfaces. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 6:13–26, 1984.
- [TK92] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: a factorization method. *International Journal of Computer Vision (IJCV)*, 9(2):137–154, 1992.
- [TM97] P. H. S. Torr and D. W. Murray. The Development and Comparison of Robust Methods for Estimating the Fundamental Matrix. *International Journal of Computer Vision*, 24(3):271–300, 1997.
- [TMHF00] Bill Triggs, Philip F. McLauchlan, Richard I. Hartley, and Andrew W. Fitzgibbon. Bundle Adjustment – A Modern Synthesis. In Springer Verlag, editor, *Vision Algorithms: Theory and Practice*, volume 1883, pages 298–372, 2000.
- [Tor95] Phil Torr. *Motion Segmentation and Outliers Detection*. PhD thesis, Department of Engineering Science, University of Oxford, 1995.
- [Tri] ARAMIS[®], Trilion Quality Systems, <http://www.trilion.com>.
- [TZ97] P. H. S. Torr and A. Zisserman. Robust Parameterization and Computation of the Trifocal Tensor. *Image and Vision Computing*, 15(8):591–605, Aug 1997.

-
- [VL93] T. Viéville and Q. Luong. Motion of points and lines in the uncalibrated case. Technical Report RR-2054, INRIA, 1993.
- [VL01] E. Vincent and E. Laganière. Matching Features Points in Stereo Pairs: A Comparative Study of Some Matching Strategies. *Machine Graphics & Vision*, 10(3):237–259, 2001.
- [VLBB⁺01] F. Vignon, G. Le Besnerais, D. Boivin, J.-L. Pouchou, and L. Quan. 3D reconstruction from scanning electron microscopy using stereovision and self-calibration. In *Physics in Signal and Image Processing*, Marseille (France), Jan 2001.
- [WCH92] J. Weng, P. Cohen, and M. Herniou. Camera Calibration with Distortion Models and Accuracy Evaluation. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 14, pages 965–980, 1992.
- [Zha96a] Zhengyou Zhang. A New Multistage Approach to Motion and Structure Estimation: From Essential Parameters to Euclidean Motion via Fundamental Matrix. Technical Report RR-2910, INRIA, Jun 1996.
- [Zha96b] Zhengyou Zhang. Determining the Epipolar Geometry and its Uncertainty: A Review. Technical Report RR-2927, INRIA, Jul 1996.
- [Zha98a] Zhengyou Zhang. A Flexible New Technique for Camera Calibration. Technical Report MSR-TR-98-71, Microsoft Research, Dec 1998.
- [Zha98b] Zhengyou Zhang. A New Multistage Approach to Motion and Structure Estimation by Gradually Enforcing Geometric Constraints. In *Asian Conference on Computer Vision (ACCV'98)*, pages 567–574, Hong Kong, Jan 1998.

Accurate 3D Shape and Displacement Measurement using a Scanning Electron Microscope

With the current development of nano-technology, there exists an increasing demand for three-dimensional shape and deformation measurements at this reduced-length scale in the field of materials research. Images acquired by Scanning Electron Microscope (SEM) systems coupled with analysis by Digital Image Correlation (DIC) is an interesting combination for development of a high magnification measurement system. However, a SEM is designed for visualization, not for metrological studies, and the application of DIC to the micro- or nano-scale with such a system faces the challenges of calibrating the imaging system and correcting the spatially-varying and time-varying distortions in order to obtain accurate measurements. Moreover, the SEM provides only a single sensor and recovering 3D information is not possible with the classical stereo-vision approach. But the specimen being mounted on the mobile SEM stage, images can be acquired from multiple viewpoints and 3D reconstruction is possible using the principle of videogrammetry for recovering the unknown rigid-body motions undergone by the specimen.

The dissertation emphasizes the new calibration methodology that has been developed because it is a major contribution for the accuracy of 3D shape and deformation measurements at reduced-length scale. It proves that, unlike previous works, image drift and distortion must be taken into account if accurate measurements are to be made with such a system. Necessary background and required theoretical knowledge for the 3D shape measurement using videogrammetry and for in-plane and out-of-plane deformation measurement are presented in details as well. In order to validate our work and demonstrate in particular the obtained measurement accuracy, experimental results resulting from different applications are presented throughout the different chapters. At last, a software gathering different computer vision applications has been developed.

Keywords: Scanning Electron Microscope, Imaging System Calibration, Distortion Correction, Drift Correction, Videogrammetry, 3D Shape Measurement, Strain Measurement, Experimental Mechanics.

Mesure précise de formes et de déformations 3D d'objets observés au Microscope Électronique à Balayage (MEB)

Avec le développement actuel des nano-technologies, la demande en matière d'étude du comportement des matériaux à des échelles micro ou nanoscopique ne cesse d'augmenter. Pour la mesure de forme ou de déformation tridimensionnelles à ces échelles de grandeur, l'acquisition d'images à partir d'un Microscope Électronique à Balayage (MEB) couplée à l'analyse par corrélation d'images numériques s'est avérée une technique intéressante. Cependant, un MEB est un outil conçu essentiellement pour de la visualisation et son utilisation pour des mesures tridimensionnelles précises pose un certain nombre de difficultés comme par exemple le calibrage du système et la correction des fortes distorsions (spatiales et temporelles) présentes dans les images. De plus, le MEB ne possède qu'un seul capteur et les informations tridimensionnelles souhaitées ne peuvent pas être obtenues par une approche classique de type stéréovision. Cependant, l'échantillon à analyser étant monté sur un support orientable, des images peuvent être acquises sous différents points de vue, ce qui permet une reconstruction tridimensionnelle en utilisant le principe de vidéogrammétrie pour retrouver à partir des seules images les mouvements inconnus du porte-échantillon.

La thèse met l'accent sur la nouvelle technique de calibrage et de correction des distorsions développée car c'est une contribution majeure pour la précision de la mesure de forme et de déformations 3D aux échelles de grandeur étudiées. Elle prouve que, contrairement aux travaux précédents, la prise en compte de la dérive temporelle et des distorsions spatiales d'images est indispensable pour obtenir une précision de mesure suffisante. Les principes permettant la mesure de forme par vidéogrammétrie et le calcul de déformations 2D et 3D sont aussi présentés en détails. Dans le but de valider nos travaux et démontrer en particulier la précision de mesure obtenue, des résultats expérimentaux issus de différentes applications sont présentés tout au long de la thèse. Enfin, un logiciel rassemblant différentes applications de vision par ordinateur a été développé.

Mots clés : Microscope Électronique à Balayage, Calibrage d'imageurs, Correction de distorsion, Correction de dérive, Vidéogrammétrie, Mesure de forme, Mesure de déformation, Photomécanique.