



Università
Ca' Foscari
Venezia

**Scuola Dottorale di Ateneo
Graduate School**

**Dottorato di ricerca
in Informatica
Ciclo 27
Anno di discussione 2015**

***High-Accuracy Camera Calibration and Scene
Acquisition***

**SETTORE SCIENTIFICO DISCIPLINARE DI AFFERENZA: INF/01
Tesi di Dottorato di Filippo Bergamasco, matricola 820576**

Coordinatore del Dottorato

Prof. Riccardo Focardi

Tutore del Dottorando

Prof. Andrea Torsello

UNIVERSITÀ CA' FOSCARI VENEZIA
DOTTORATO DI RICERCA IN INFORMATICA, 27° CICLO
(A.A. 2011/2012 – 2013/2014)

High-Accuracy Camera Calibration and Scene Acquisition

SETTORE SCIENTIFICO DISCIPLINARE DI AFFERENZA: INF/01

TESI DI DOTTORATO DI FILIPPO BERGAMASCO
MATR. 820576

TUTORE DEL DOTTORANDO

Andrea Torsello

COORDINATORE DEL DOTTORATO

Riccardo Focardi

October, 2014

Author's Web Page: <http://www.dsi.unive.it/~bergamasco>

Author's e-mail: bergamasco@dsi.unive.it

Author's address:

Dipartimento di Scienze Ambientali, Informatica e Statistica
Università Ca' Foscari Venezia
Via Torino, 155
30172 Venezia Mestre – Italia
tel. +39 041 2348465
fax. +39 041 2348419
web: <http://www.dais.unive.it>

*To my beloved Francesca,
for her unconditional and continuous
support over the years.
To my family, Andrea Torsello and the whole cv::lab group*

- Filippo Bergamasco

Abstract

In this thesis we present some novel approaches in the field of camera calibration and high-accuracy scene acquisition. The first part is devoted to the camera calibration problem exploiting targets composed by circular features. Specifically, we start by improving some previous work on a family of fiducial markers which are leveraged to be used as calibration targets to recover both extrinsic and intrinsic camera parameters. Then, by using the same geometric concepts developed for the markers, we present a method to calibrate a pinhole camera by observing a set of generic coplanar circles.

In the second part we move our attention to unconstrained (non-pinhole) camera models. We begin asking ourselves if such models can be effectively applied also to quasi-central cameras and present a powerful calibration technique that exploit active targets to estimate the huge number of parameters required. Then, we apply a similar method to calibrate the projector of a structured-light system during the range-map acquisition process to improve both the accuracy and coverage. Finally, we propose a way to lower the complexity of a complete unconstrained model toward a pinhole configuration but allowing a complete generic distortion map.

In the last part we study two different scene acquisition problems, namely: Accurate 3D shape reconstruction and object material recovery. In the former, we propose a novel visual-inspection device for the dimensional assessment of metallic pipe intakes. In the latter, we formulate a total-variation regularized optimization approach for the simultaneous recovery of the optical flow and the dichromatic coefficients of a scene by analyzing two subsequent frames.

Sommario

In questa tesi sono proposti nuovi approcci nel campo della calibrazione di videocamere digitali e ricostruzione accurata della scena. La prima parte è dedicata al problema della calibrazione, affrontato sfruttando target composti da pattern di features circolari. Nello specifico vengono dapprima approfonditi degli approcci precedentemente pubblicati dall'autore su famiglie di fiducial markers basati su schemi di punti circolari e studiato il loro utilizzo come target di calibrazione. Successivamente, sfruttando alcuni concetti di geometria proiettiva introdotti per i markers, viene discusso un metodo per la calibrazione di videocamere pinhole attraverso l'analisi di insiemi di generici cerchi coplanari.

Nella seconda parte della tesi vengono approfondite tematiche riguardanti la calibrazione di videocamere assumendo un modello a raggi liberi non pinhole. La trattazione inizia ponendosi il quesito se tali modelli possano efficacemente essere sfruttati (in termini di accuratezza e facilità di calibrazione) anche nei contesti in cui il modello pinhole con distorsione radiale è globalmente considerato ottimale. Nello studio della questione viene proposta una tecnica di calibrazione basata su un'ottimizzazione alternata di un funzionale che permette la stima dei parametri che descrivono ciascun raggio sfruttando corrispondenze ricavate da target attivi. Successivamente, la stessa tecnica viene utilizzata per calibrare il proiettore di un sistema di scansione a luce strutturata in contemporanea con il processo di acquisizione. Infine, viene proposto un modello a raggi liberi vincolati a passare per un unico centro ottico, fornendo quindi una metodologia per creare camere pinhole virtuali dotate di una mappa di distorsione completamente non vincolata. L'approccio combina la potenza del modello pinhole (offrendo tutti gli strumenti matematici di geometria proiettiva) con l'accuratezza di un modello a raggi liberi.

Infine, nell'ultima parte vengono affrontati due diversi problemi di acquisizione. Nel primo, viene proposto un sistema industriale di misurazione di imboccature elittiche di tubature. Nel secondo viene presentato un algoritmo per la stima simultanea dei parametri del modello dicromatico e dell'optical flow a partire da dati multi-spettrali acquisiti da una scena.

Contents

Preface	xv
Published Papers	xvii
1 Introduction	1
1.1 Camera Modelling and Calibration	2
1.2 Multi-Spectral imaging	4
2 Related Work	7
2.1 The Pinhole Camera Model	7
2.1.1 Lens Distortion	9
2.1.2 Calibration techniques	12
2.1.3 Projective planes, conics and ellipse fitting	14
2.2 Camera pose estimation	16
2.2.1 Fiducial Markers	17
2.3 Non-Pinhole Models	19
2.3.1 Wide-angle, Fish eye and Catadioptric Cameras	19
2.3.2 Plenoptic Cameras	20
2.3.3 Raxel-based Unconstrained Imaging Models	21
2.4 Optical Flow Estimation	22
2.5 Reflectance Modelling	24
I Calibrating with circular features	27
3 A Projective Invariants Based Fiducial Marker Design	29
3.1 Introduction	30
3.2 Image-Space Fiducial Markers	30
3.2.1 Projective invariants	30
3.2.2 Marker Detection and Recognition	32
3.2.3 Estimation of the Camera Pose	34
3.3 Experimental Validation	35
3.3.1 Accuracy and Baseline Comparisons	36
3.3.2 Resilience to Occlusion and False Ellipses	37
3.3.3 Performance Evaluation	37
3.3.4 Behavior on Real Videos	39
3.3.5 Using Pi-Tag for camera calibration	39

3.3.6	Contactless measurements	42
3.3.7	Scalability over the number of markers	43
3.3.8	Registration of 3D surfaces	46
3.3.9	Applications in Projected Augmented Reality	46
3.4	Conclusions	50
4	Robust Fiducial Marker	
	Based on Cyclic Codes	51
4.1	Introduction	52
4.2	Rings of UNconnected Ellipses	52
4.2.1	Candidate selection with a calibrated camera	52
4.2.2	Dealing with the uncalibrated case	57
4.2.3	Marker Recognition and Coding Strategies	60
4.3	Experimental Validation	62
4.3.1	Accuracy and Baseline Comparisons	64
4.3.2	Resilience to Occlusion and Illumination	64
4.3.3	RUNE Tags for camera calibration	65
4.3.4	Mono vs. Stereo Pose Estimation	66
4.3.5	Performance Evaluation	68
4.3.6	Shortcomings and Limitations	68
4.4	Conclusions	68
5	Camera Calibration	
	from Coplanar Circles	71
5.1	Introduction	72
5.1.1	Our approach	73
5.1.2	Ellipse detection and refinement	74
5.2	Selecting Coplanar Circles with a Non Cooperative Game	74
5.3	Camera parameters optimization	75
5.3.1	Problem formulation	76
5.3.2	Energy minimization	77
5.4	Experimental Evaluation	77
5.5	Conclusion	81
II	Model-free camera calibration	83
6	Can an Unconstrained Imaging	
	Model be Effectively Used	
	for Pinhole Cameras?	85
6.1	Introduction	86
6.2	Imaging Model and Calibration	86
6.2.1	Least Squares Formulation	87

6.2.2	Ray Calibration	89
6.2.3	Estimation of the Poses	90
6.2.4	Accounting for Refraction	90
6.3	Working with the Unconstrained Camera	92
6.4	Rays interpolation	93
6.4.1	Rays manifold interpolation function	94
6.4.2	Selecting the interpolation data	95
6.5	Experimental Evaluation	96
6.6	Discussion	100
7	High-Coverage Scanning trough	
	Online Projector Calibration	103
7.1	Introduction	104
7.2	High-Coverage 3D Scanning	106
7.2.1	Online Projector Calibration	106
7.2.2	Outliers Filtering	107
7.3	Experimental Evaluation	108
7.3.1	Enhanced Coverage	109
7.3.2	Reconstruction accuracy	111
7.3.3	Surface Repeatability	112
7.3.4	Planarity	113
7.4	Conclusion	113
8	Non-Parametric Lens Distortion Estimation for Pinhole Cameras	115
8.1	Introduction	116
8.2	Unconstrained Distortion Map for Central Camera Calibration	116
8.2.1	Single Camera calibration	117
8.2.2	Dealing with Stereo Cameras	124
8.3	Experimental Section	125
8.3.1	Image Undistortion	126
8.3.2	3D Measurement	127
8.4	Conclusions	127
III	Reconstruction and Measurement Applications	129
9	Robust Multi-Camera	
	3D Ellipse Fitting	131
9.1	Introduction	132
9.2	Multiple Camera Ellipse Fitting	133
9.2.1	Parameterization of the 3D Ellipse	134
9.2.2	Energy Function over the Image	135
9.2.3	Gradient of the Energy Function	136
9.3	Experimental evaluation	138

9.3.1 Synthetic Experiments	140
9.3.2 Real World Application	141
9.4 GPU-based Implementation	143
9.5 Conclusions	144
10 Simultaneous Optical Flow and Dichromatic Parameter Recovery	145
10.1 Introduction	146
10.2 Multi-view Dichromatic Parameter Recovery	146
10.2.1 Multi-Spectral Imaging and the Dichromatic Model	146
10.2.2 Optical Flow and Reflectance Coherency	147
10.2.3 Total Variation Regularization	148
10.2.4 Multi-view dichromatic functional	149
10.3 Minimization Process	149
10.3.1 Initialization	151
10.3.2 Effect of the regularization terms	152
10.4 Experiments	154
10.5 Conclusions	158
11 Conclusions	159
11.1 Future Work	160
Bibliography	163

List of Figures

1.1	A visual representation of the different components of a camera imaging model. The camera pose parameters relate the camera and world reference frame. Furthermore, camera lenses project light rays coming from the scene to the sensor image plane to produce the final output image.	3
1.2	A representation of the visible light spectrum varying the wavelength (expressed in nano-meters).	4
1.3	Data cube representation of a multi-spectral acquisition device.	5
2.1	Left: Path followed by light rays entering the camera in a pinhole model. Right: The pinhole model conventionally used with the optical center placed behind the image plane. Note how the projection is substantially the same but is not vertical-mirrored.	8
2.2	The projection process of a pinhole camera. First, a 3D point \mathbf{M} is transformed through the rigid motion \mathbf{R}, \vec{T} . Then, the transformed point is projected so that \mathbf{m} is the intersection between the line connecting \mathbf{o} and \mathbf{M} and the image plane.	9
2.3	Different types of radial distortions. Left: Original undistorted image. Center: positive (barrel) distortion. Right: negative (pincushion) distortion.	11
2.4	Some examples of fiducial markers that differ both for the detection technique and for the pattern used for recognition. In the first two, detection happens by finding ellipses and the coding is respectively held by the color of the rings in (a) and by the appearance of the sectors in (b). The black square border enables detection in (c) and (d), but while AR-Toolkit uses image correlation to differentiate markers, ARTag relies in error-correcting binary codes. Finally, in (e) and (f) we show two classes of RUNE-Tags fiducial markers described in Chapter 4.	18
2.5	Example of a fish eye lens (Left) with a sample picture of the result obtainable (Right). Note the severe distortion that cause straight lines to appear curved.	19
2.6	Left: A schematic representation of the Lytro™ camera. Right: The scene projection process on a plenoptic camera based on array of micro lenses. A standard camera lens focuses the scene onto the micro lenses array that act like multiple small pinhole cameras converging the captured rays to the device sensor. This way, the whole light field entering the main lens can be acquired.	20

2.7	An example of optical flow estimation from two images (Left and Center) taken from a moving camera. At each point of the first image is associated a displacement (flow) vector to the corresponding point on the second.	22
3.1	The cross-ratio of four collinear points is invariant to projective transformations. $cr(A, B, C, D) = cr(A', B', C', D')$	31
3.2	Steps of the marker detection process: in (a) a good candidate for a side is found by iterating through all the point pairs ($O(n^2)$). In (b) another connected side is searched for and, if found, the resulting angular ordering is used to label the corners found ($O(n)$). Note that the labeling is unambiguous since the corner i is associated with the lowest cross ratio. Finally, in (c) the marker is completed (if possible) by finding the missing corner among all the remaining dots. (image best viewed in colors)	32
3.3	Evaluation of the accuracy of camera pose estimation with respect to different scene conditions. The first row plots the angular error as a function of view angle and Gaussian blur respectively, while the second row plots the effects of Gaussian noise (left) and illumination gradient (right, measured in gray values per image pixel). The proposed method is tested both with and without refinement. Comparisons are made with ARToolkit and ARToolkit Plus.	34
3.4	Some examples of artificial noise used for synthetic evaluation. The artificial noise is, respectively, light Gaussian noise at grazing view angle (first column), blur (second column), strong Gaussian noise (third column) and illumination gradient (fourth column). The tested markers shown are ARToolkit Plus (first row) and Pi-Tag (second row).	35
3.5	Left (a): Evaluation of the accuracy of the estimated camera pose when some dot of the marker are occluded (note that if more than 5 dots are missing the marker is not detected). Right (b): Evaluation of the number of false positive markers detected as a function of the number of false ellipses introduced in the scene and the threshold applied to the cross-ratio.	37
3.6	Left (a): Evaluation of the detection and recognition time for the proposed marker as random ellipses are artificially added to the scene. Right (b): Evaluation of the recognition rate achieved on a real video of about ten minutes in length, with respect to different thresholds applied to the cross-ratio.	38
3.7	Recognition fails when the marker is angled and far from the camera as the ellipses detectors cannot detect the circular features.	38
3.8	Some examples of the behaviour in real videos: In (a) the marker is not occluded and all the dots contribute to the pose estimation. In (b) the marker is recognized even if a partial occlusion happens. In (c) the marker cannot be detected as the occlusion is too severe and not enough ellipses are visible.	39

3.9	Evaluation of the quality of mono and stereo calibration obtained using Pi-Tags as fiducial markers.	40
3.10	Performance of the proposed fiducial marker as a tool for image-based measurement.	41
3.11	Analysis of the measurement error committed with respect to different positions of the marker pair.	41
3.12	Cross ratios measured in a real video sequence.	43
3.13	Relation between recognition margin and cross-ratio separation among markers.	44
3.14	Examples of surface reconstructions obtained by acquiring several ranges with a structured-light scanner and by using Pi-Tag markers to set a common reference (image best viewed in color).	45
3.15	Actual setup and examples of usage by moving the controller above the printed map.	46
3.16	A schematic representation of the setup.	47
3.17	Geometric relation between the entities involved in the projector calibration procedure.	49
4.1	Our proposed marker design divided into its functional parts. An instance of a RUNE-129 with 3 levels is displayed.	53
4.2	Number of maximum steps required for ellipse testing.	54
4.3	The four possible camera orientations that transform an observed ellipse into a circle	55
4.4	Steps of the ring detection: in (a) the feasible view directions are evaluated for each ellipse (with complexity $O(n)$), in (b) for each compatible pair of ellipses the feasible rings are estimated (with complexity $O(n^2)$), in (c) the dot votes are counted, the code is recovered and the best candidate ring is accepted (figure best viewed in color).	56
4.5	Estimated normals orientation in spherical coordinates of three coplanar ellipses spanning positive (Left) and negative (Right) focal length values. Note how one of the two possible orientations converge to a common direction while the other does the opposite.	57
4.6	A synthetic representation of the marker dots normal voting scheme used to guess an initial value of the camera focal length. Left: RUNE-129 markers rendered by a virtual camera with known focal length and principal point. Center: the normal accumulator in spherical coordinates. Right: Focal length distribution of the bins. See the text for a complete discussion on the voting procedure.	59
4.7	Evaluation of the accuracy in the camera pose estimation with respect to different scene conditions. Examples of the detected features are shown for RUNE-129 (first image column) and ARToolkitPlus (second image column).	63

4.8	Some examples of behaviour in real videos with occlusion. In (a) and (b) an object is placed inside the marker and the setup is rotated. In (c) and (d) the pose is recovered after medium and severe occlusion.	63
4.9	Evaluation of the accuracy in the camera pose estimation of RUNE-Tag with respect to occlusion (left column) and illumination gradient (right column).	64
4.10	Evaluation of the recognition time respectively when adding artificial false ellipses in the scene (left column) and with several markers (right column).	65
4.11	Recognition rate of the two proposed marker configurations with respect to the percentage of area occluded.	65
4.12	Accuracy of camera calibration while using a single RUNE-129 as a dot-based calibration target. Camera poses has been divided into 3 groups based on the maximum angle between the camera z -axis and the marker plane. A random subset of photos is used to test the calibration varying the number of target exposures. In all the experiments we achieve a good accuracy with a decreasing st.dev. when increasing the number of photos.	67
4.13	Comparison between the pose accuracy for a single or stereo camera setup. Left: distance between two jointly moving markers as a function of the angle with respect to the first camera. Right: Angle around the marker plane normal as estimated by the first camera versus the stereo setup. Ideally, all the measures should lie on the 45 degrees red line.	67
4.14	Recognition fails when the marker is angled and far away from the camera and the ellipses blends together.	69
5.1	A possible calibration target composed by some coplanar circles.	73
5.2	Left: Score obtained by a set of games played for different candidate focal length spanning around the correct known value of 1500. Note the clear maximum obtained around such value. Right: Final population after two non-cooperative games with the correct focal length value ($f=1500$) and a wrong value ($f=1000$). In the correct case, almost all winning strategies will exhibit the same value.	75
5.3	Some qualitative calibration examples on real world scenes. Left: Original images. Right: Rectified images obtained from the estimated camera intrinsics and orientation.	79
5.4	Focal length estimation error after the optimization varying the initial focal length guess.	80
5.5	Focal length estimation error after the optimization varying the initial optical center guess.	80
5.6	Estimated focal length (Left), optical center x (Center) and optical center y (Right) with respect of the number of poses (Top row) and noise (Bottom row). Ground truth is indicated with a dashed line.	81

6.1	Schema of the general camera model and calibration target described in this chapter. Note that the Mahalanobis distance between observed and expected code is equal to the 3D distance of observed code to the ray. . . .	88
6.2	Effect of refraction correction for different values of the refraction parameters.	91
6.3	Manifold interpolation of free rays.	93
6.4	Root mean squared error between expected and observed codes as a function of the number of iterations of the calibration algorithm. The top plot shows the error averaged over all the pixels and poses, while the bottom pictures show for each pixel the error averaged over all the poses at iteration 0 (pinhole model), 2, and 21.	97
6.5	Comparison of the error obtained with the pinhole model calibrated with a chessboard and dense target, and with our calibration approach for the unconstrained model.	98
6.6	Scatter plot of the measured distances between pairs of points taken in the top part (left) and bottom part (right) of the target. The points where 620 target pixels apart.	98
6.7	Spatial distribution of the local pinholes, i.e., of the closet point to the rays in a local neighborhood. In a pinhole model all the points would coincide. For display purposes the points are sub-sampled.	101
7.1	The incomplete coverage problem that affects many structured light systems. See the text for details. (image best viewed in color)	105
7.2	The bundles of rays that can be obtained after calibration of the projector using the reconstructed 3D points. In the first image we adopted the pinhole+distortion model. The second and third image show the results obtained using the unconstrained model respectively with and without outlier correction. Note that the pinhole model is able to calibrate all the rays, while the unconstrained model can be populated only by the rays that hit the scanned surface, thus they are a bit less. Also note that all the miscalibrated rays have (apparently) disappeared after outlier removal. . .	106
7.3	Coverage difference between the baseline (top row) and the unconstrained method (bottom row) for some different subjects.	108
7.4	Scanner head with 2 calibrated cameras and an uncalibrated projector. . .	109
7.5	Increment in the coverage with respect to the number of scans.	110
7.6	Accuracy of the reconstruction with respect to the baseline method. The close-ups on the left part of the figure show a detail of the reconstruction obtained respectively with the baseline, unconstrained and pinhole methods.	110
7.7	Repeatability of the reconstruction for different scans of the same subject. On the right part of the figure we show some slices from the acquired meshes to illustrate the alignment between subsequent scans respectively with the baseline, pinhole and unconstrained methods.	111

7.8	Coherence of the reconstructed surface with a planar reference target. . .	112
8.1	Schema of the optimized camera model involving the optical center o , the ray direction, the observed and expected code and a target pose	117
8.2	RMS of the error between the observed and the expected code for each $r_{(u,v)}$ at the first (left image) and last (right image) iteration of the optimization process of rays, optical center, and poses.	120
8.3	Spatial configuration of the optimized camera. Camera z-axis is aligned with the plane ν_ϕ . The intersection between all rays and the plane inherits the lattice topology of the camera sensor. The points lattice (in black) on the image plane is resampled in a uniform grid (in red) to create the undistortion function.	121
8.4	The effect of the observed codes filtering step displayed by accumulating the value $E(u, v)$ among all the poses s . Left: original data may contain clear outliers near the boundary of the target. Right: after the filter almost all the spikes are no more visible.	124
8.5	Left plot shows a comparison of our non-parametric distortion model compared with Fitzgibbon's rational model. For a complete description of the displayed result see the text. On the right, an example of an undistorted image generated with our method.	126
8.6	In the left-most plot we show the average relative error between the expected and measured distance of two triangulated points placed at a certain known distance. In the right we show an example of a stereo-rectified image pair.	127
8.7	Reconstructed range-map triangulated from the OpenCV calibration and rectification (Left) and our proposed method (Right).	128
9.1	Schematic representation of a multi-camera system for industrial in-line pipes inspection.	132
9.2	Evaluation of the accuracy of the proposed method with respect to different noise sources. The metric adopted is the relative error between the minor axis of the ground truth and of the fitted ellipse.	137
9.3	The experimental Multiple-camera imaging head.	139
9.4	Examples of images with artificial noise added. Respectively additive Gaussian noise and blur in the left image and occlusion in the right image. The red line shows the fitted ellipse.	140
9.5	Comparison between the accuracy of the initial 2D fitting and the proposed 3D optimization.	141
9.6	Quantitative assessment of the improvement in accuracy (Left) and effect of the number of views over the measure quality.	142

9.7	Comparison between the running time of the CPU and GPU-based implementations of the multiview algorithm. Times are plotted with respect to the number of pixels in the evaluated mask (i.e. size of the ellipse to be refined).	144
10.1	Sample image pair showing the effect of the priors on the regularized parameters. First row: input image pair. Second and third row: Reflectance value and gradient magnitude computed by H&RK (left) and by our approach (right). Last row: Forbenious norm of the differential of the hyperparameters A at initialization and after optimization.	153
10.2	Reflectance obtained for two sample image pairs from scene 3 (left) and 4 (right). First row: input images. Second row: cromaticity obtained with H&RK and our method. Third row: gradient magnitude of the reflectances.	154
10.3	Shading (top-row) and specular factors (bottom-row) obtained for the same sample image pairs shown in Figure 10.2.	155
10.4	Illuminant power spectra for each scene. First and second column: power spectrum for each scene image as computed by H&RK and our approach respectively. Third column: average spectrum with the standard deviation for each band.	156
10.5	Reflectance spectra and the standard deviation for each band for the pixels inside the respective color tiles.	157

List of Tables

- 10.1 RMS and Euclidean angular error for the illuminant recovered by our approach and the H&RK method for all the four scenes of our dataset. . . . 155
- 10.2 Average and standard deviation of the RMS and Euclidean angular error of the estimated reflectance inside the colored tiles shown in Figure 10.5. 157

Preface

My journey as a young researcher started almost 5 years ago. It was back in 2009 when I started working with a young but really motivated team of students, researchers and enthusiasts at Ca'Foscari University of Venice. At that time, I've been involved into the development of minor parts of a complete structured-light scanning solution for a big company in Italy.

Many things changed since that time, someone in the group left and someone new arrived. Still, the original spirit that drove us in the search of excellence, the desire of undertake new challenges and the positive inspiration that you could breath is still here. For this reason, this thesis is dedicated to you all. What you will find here is a subset of many things that I had the possibility to explore during my PhD while working on new state-of-the-art ideas and techniques in the field of camera calibration and scene reconstruction. I did my best to carefully collect and present our work into a coherent and pleasant discussion so that it could be a basis for future insights and enhancements. But...for me is far more than that. I'd like to think at this thesis as a summary of all the great times I had in the last 4 years that really influenced my passions and shaped my aspirations to what I hope gonna be a brilliant future.

“The only true voyage of discovery, the only fountain of Eternal Youth, would be not to visit strange lands but to possess other eyes, to behold the universe through the eyes of another, of a hundred others, to behold the hundred universes that each of them beholds, that each of them is.”

Marcel Proust, in “Remembrance of Things Past”

Published Papers

- [1] LUCA COSMO, ANDREA ALBARELLI, FILIPPO BERGAMASCO A low cost tracking system for position-dependent 3D visual interaction *Proceedings of the 2014 International Working Conference on Advanced Visual Interfaces - AVI '14, ACM Association for Computing Machinery*, pp. 351- 352, 2014.
- [2] FILIPPO BERGAMASCO, ANDREA ALBARELLI, EMANUELE RODOLA, ANDREA TORSELLO Can a Fully Unconstrained Imaging Model Be Applied Effectively to Central Cameras? *IEEE Conference on Computer Vision and Pattern Recognition, IEEE*, pp. 1391- 1398, 2013.
- [3] ANDREA ALBARELLI, FILIPPO BERGAMASCO, AUGUSTO CELENTANO, LUCA COSMO, ANDREA TORSELLO Using multiple sensors for reliable markerless identification through supervised learning *MACHINE VISION AND APPLICATIONS (ISSN:0932-8092)*, pp. 1539- 1554, 2013.
- [4] FILIPPO BERGAMASCO, ANDREA ALBARELLI, ANDREA TORSELLO Pi-Tag: a fast image-space marker design based on projective invariants *MACHINE VISION AND APPLICATIONS (ISSN:0932-8092)*, pp. 1295- 1310, 2013.
- [5] FILIPPO BERGAMASCO, ANDREA ALBARELLI, ANDREA TORSELLO A Practical Setup for Projection-based Augmented Maps *First International Conference on Software and Emerging Technologies for Education, Culture, Entertainment, and Commerce: New Directions in Multimedia Mobile Computing, Social Networks, Human-Computer Interaction and Communicability, Blue Herons*, pp. 13- 22., 2012.
- [6] FILIPPO BERGAMASCO, LUCA COSMO, ANDREA ALBARELLI , ANDREA TORSELLO A Robust Multi-Camera 3D Ellipse Fitting for Contactless Measurements *2nd Joint 3DIM/3DPVT Conference 3D Imaging, Modeling, Processing, Visualization, Transmission, IEEE*, pp. 168- 175., 2012.
- [7] F. BERGAMASCO, A. ALBARELLI, A. TORSELLO, M. FAVARO, P. ZANUTTIGH Pairwise Similarities for Scene Segmentation combining Color and Depth data *21st International Conference on Pattern Recognition (ICPR 2012), IEEE COMPUTER SOCIETY*, pp. 3565- 3568 , 2012.
- [8] A. ALBARELLI, F. BERGAMASCO, L. ROSSI, S. VASCON, A. TORSELLO A Stable Graph-Based Representation for Object Recognition through High-Order Matching *21st International Conference on Pattern Recognition (ICPR 2012), IEEE COMPUTER SOCIETY*, pp. 3341- 3344, 2012.

-
- [9] EMANUELE RODOLÀ, ANDREA ALBARELLI, FILIPPO BERGAMASCO, ANDREA TORSELLO A Scale Independent Selection Process for 3D Object Recognition in Cluttered Scenes *INTERNATIONAL JOURNAL OF COMPUTER VISION (ISSN:0920-5691)*, pp. 129- 145 , 2013.
- [10] ANDREA ALBARELLI, FILIPPO BERGAMASCO, ANDREA TORSELLO Rigid and Non-rigid Shape Matching for Mechanical Components Retrieval *11th IFIP TC 8 International Conference, CISIM 2012, Springer*, pp. 168- 179 , 2012.
- [11] FILIPPO BERGAMASCO, ANDREA ALBARELLI, ANDREA TORSELLO A graph-based technique for semi-supervised segmentation of 3D surfaces *PATTERN RECOGNITION LETTERS (ISSN:0167-8655)*, pp. 2057- 2064 , 2012.
- [12] E. RODOLA, ALEX M. BRONSTEIN, A. ALBARELLI, F. BERGAMASCO, A. TORSELLO A Game-Theoretic Approach to Deformable Shape Matching *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2012)*, 2012.
- [13] F. LECKLER, F. ARDHUIN, A. BENETAZZO, F. FEDELE, F. BERGAMASCO, V. DULOV Space-time properties of wind-waves: a new look at directional wave distributions *EGU General Assembly 2014, At Vienna, Austria, Volume: Vol. 16, EGU2014-13508-2*, 2014.
- [14] A. BENETAZZO, F. BERGAMASCO, F. BARBARIOL, A. TORSELLO, S. CARNIEL, M. SCLAVO Towards an Operational Stereo System for Directional Wave Measurements from Moving Platforms *SME 2014 33rd International Conference on Ocean, Offshore and Arctic Engineering (OMAE2014)*, 2014.

1

Introduction

In the recent past, the optical acquisition and measurement of a scene was only viable for research labs or professionals that could afford to invest in expensive and difficult to handle high-end hardware. However, due to both technological advances and increased market demand, this scenario has been altered significantly. Indeed, semi-professional imaging devices can be found at the same price level of a standard workstation, widely available software stacks can be used to obtain reasonable results even with low-end hardware and, finally, a constant dramatic improvement of computing power now allows the addressing of complex (possibly under-constrained) problems via powerful non-linear optimization approaches.

Moreover, we assisted an advance in digital imaging devices in terms of resolution, sensitivity, dynamic range and cost. All these factors combined are driving the adoption of vision-based solutions to a broad range of applications, spanning from high precision photogrammetry and 3D reconstruction to video surveillance and home entertainment. Just to cite few examples, it is now common for the manufacturing sector to arrange one or more 3D sensors to assess the products dimensional characteristics [34, 127, 17]. Besides, research areas traditionally not devoted to the usage of optical instruments are now able to acquire data with unprecedented resolution and accuracy [30, 94]. Finally, quite remarkable is the recent success of innovative controllers for the video game industry like Microsoft's Kinect™ or Sony's Playstation Move™ which, among many other innovations like gesture-controlled televisions, are leading the mainstream market to a pervasive adoption of environment-sensing technologies. However, an accurate modelling of a given imaging machinery and the study of advanced reconstruction algorithms is still a critical requirement to exploit the full potential of today's hardware.

In this thesis I present some selected works performed during my PhD studies mostly devoted to the problem of camera calibration and high-accuracy scene acquisition. The camera calibration problem is firstly discussed for the pinhole case both in terms of extrinsic and intrinsic parameters estimation exploiting the properties of circular features that are either specially crafted (in form of fiducial markers) or assumed to be present (as sets of coplanar circles) in a scene. The former is discussed in Chapter 3 and 4 while the latter is presented in Chapter 5.

Then, we move our attention to a more powerful unconstrained camera model in

which each single ray entering the camera is independently described. Specifically, in Chapter 6 we start by proposing an effective way to calibrate such model and experimentally demonstrate that it can be used with great advantages even for quasi-pinhole cameras. Subsequently, in Chapter 7 we apply the newly introduced calibration method to a structured-light 3D scanning setup by means of an on-line calibration of the projector light path that happens simultaneously with the acquisition. Finally, in Chapter 8 we propose a way to lower the complexity of a complete unconstrained model toward a pinhole configuration but allowing a complete generic distortion mapping. This would act as a powerful bridge between the simplicity of the central projection assumption and the flexibility of the unconstrained raxel-based model.

The scene acquisition problem will be discussed both in terms of industry-grade 3D elliptical shape reconstruction (Chapter 9) and surface characteristics recovery, i.e. reflectance, shading and illuminant spectrum estimation (Chapter 10). In the former, we propose a novel visual-inspection device for the dimensional assessment of metallic pipe intakes. In the latter, we formulate a total-variation regularized optimization approach for the simultaneous recovery of the optical flow and the dichromatic coefficients of a scene by analyzing two subsequent frames.

Before proceeding with the discussion of the main topics of this thesis, we reserved the following sections to introduce some fundamental concepts and tools. This would be useful to both the practitioner and experienced reader either to fully understand the subsequent chapters or to clarify some notations that will be used throughout this thesis. Furthermore, in Chapter 2 we aim to give a comprehensive review of the existing literature to better frame the novelty of this thesis with respect with the current state-of-the-art in each respective fields.

1.1 Camera Modelling and Calibration

While being conceptually as simple as the first stenopeic hole, typical imaging cameras are equipped with complex lens setups, sensible and dense CCD or CMOS sensors, high speed ADC converters and, sometimes, even advanced band-pass filters to acquire data in a broad range of spectral bands. As a consequence, the ability to exploit such features at best implies a reasoned and physically accurate description of their inner working into appropriate mathematical models.

Since the beginning of computer vision, two joint fundamental problems attracted the attention of the researchers worldwide. First, the definition of mathematical models that can properly describe the camera image formation process and, second, the parameters estimation of such model in a practical yet effective way. At its basic level, a digital camera is composed by an array of light-sensitive cells that transform the amount of photons hitting their surface (at certain frequencies) into electric signals. In most of the cases, such basic photons detectors are disposed in a grid producing a collection of numerical values arranged into one or more (in case of multi-spectral devices) matrices. Furthermore, to produce sharp and meaningful images, a set of dif-

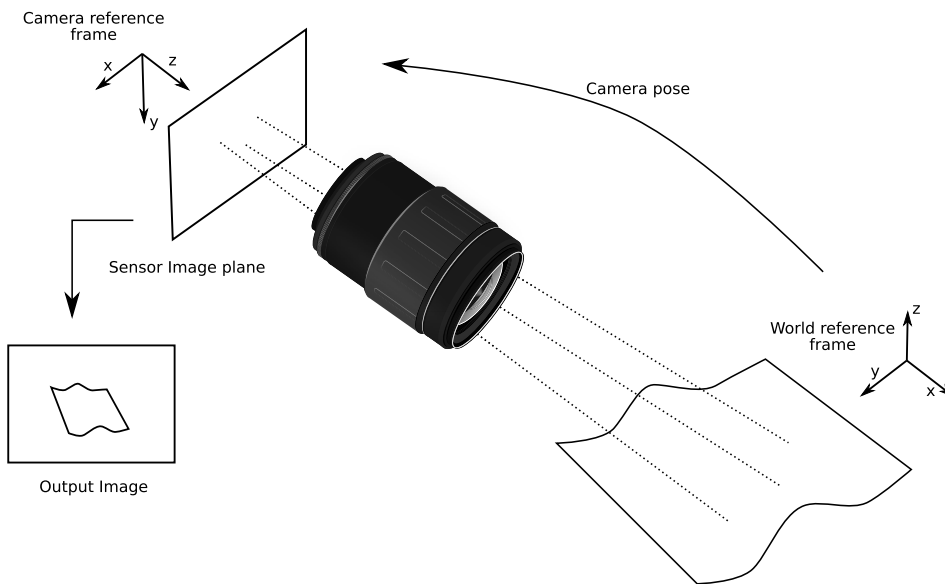


Figure 1.1: A visual representation of the different components of a camera imaging model. The camera pose parameters relate the camera and world reference frame. Furthermore, camera lenses project light rays coming from the scene to the sensor image plane to produce the final output image.

ferent shaped lenses are adopted to control the path of the light rays coming from the scene and hitting the sensors.

When a camera is used to observe a three-dimensional scene composed by some geometrical primitives being either points, lines or surfaces, all the light rays collected produce a two-dimensional representation of such scene onto the sensor image plane (Fig. 1.1). In this sense, camera calibration deals with the problem of estimating such 3D to 2D function that drives the image formation process of a scene. Obviously, this mapping is not injective causing a loss of information during the projection. However, when this mapping is known (i.e. calibrated), multiple cameras can be simultaneously used to recover the structure of a scene (i.e. 3D coordinates of points and primitives) given the image planes projections [86].

Independently of the specific camera model assumed, the mapping function can be divided into two different steps. In the first, the 3D primitives are transformed from the world to the camera reference frame by means of a rigid-motion. The parameters controlling such roto-translation, defining the *camera pose* with respect to the scene up to 6-degrees of freedom, are called *Extrinsic Parameters*. Usually, a 3×3 rotation matrix \mathbf{R} and a $\vec{T} \in \mathbb{R}^3$ translation vector are used thus requiring the estimation of 12 different (constrained) parameters. In the second step, the 3D points now lying in the camera reference frame are projected into the image plane. This mapping, depending

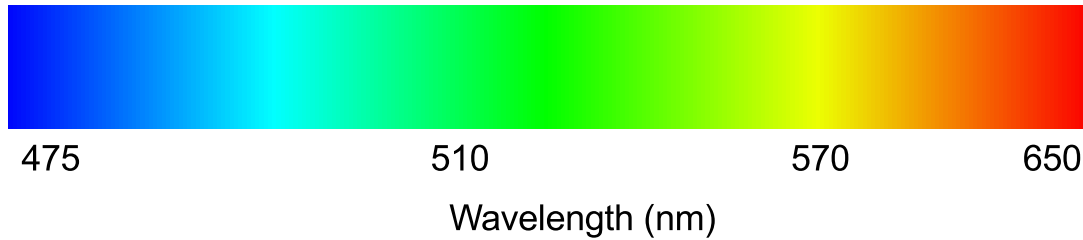


Figure 1.2: A representation of the visible light spectrum varying the wavelength (expressed in nano-meters).

only on the internal geometry and optical behaviour of the camera, is controlled by a set of *Intrinsic Parameters* or *Internal Camera Parameters* whose number and meaning may vary with respect to the model.

1.2 Multi-Spectral imaging

Cameras discussed so far were simply described as light photons collectors. However, it's well known that light can either behave as particles (i.e. photons) or electromagnetic waves [79]. From the waves point of view, the camera sensor is measuring the amplitude of electromagnetic waves whose wavelength vary around the so called visible spectrum (Fig. 1.2).

Usually, for gray-scale cameras no distinction is made at all to the frequency, integrating indifferently over time on a certain spectral range. However, if we discriminate with respect to the wavelengths, the output of our imaging device would become a “cube”, where the two dimensions spanning the spatial arrangement of light sensitive cells are the usual coordinates of the image lattice pixels whereas the third dimension corresponds to the wavelength (Fig.1.3). Consequently, we can think at each image pixel located at coordinates (u, v) as a vector in \mathbb{R}^n whose components represent the intensity (*Radiance*) of the light radiation collected at coordinates (u, v) in the image plane for a specific wavelength $\lambda \in \{\lambda_1 \dots \lambda_n\}$.

A common type of multi-spectral imaging device is the RGB color camera that can be thought as operating on just 3 band ranges spanning around the wavelengths of 450 nm (blue), 550 nm (green) and 600 nm (red). The number of bands, the size of each wavelength interval and the amount of spectrum covered discriminates between multi-spectral and hyper-spectral imagery. The first usually refers to devices designed to cover few (8-16) broad bands, that may be discontinuous. Conversely, hyper-spectral imaging operates on dozens of narrow-bands covering a continuous range of different wavelengths.

The availability of multi-spectral data allows a multitude of different scene analy-

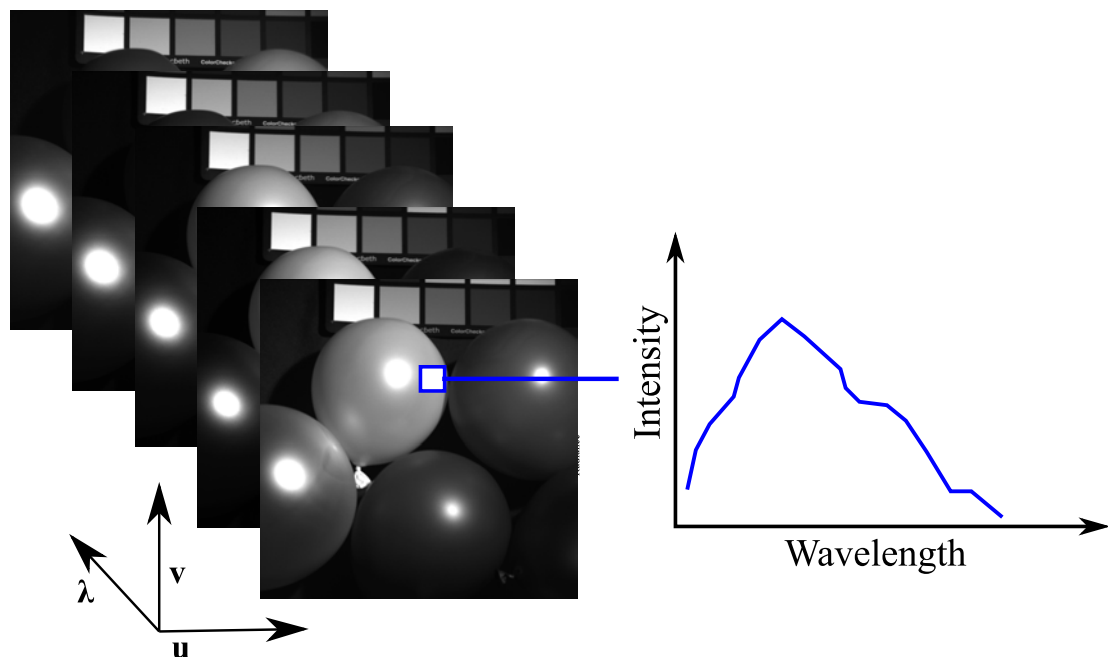


Figure 1.3: Data cube representation of a multi-spectral acquisition device.

sis applications that goes beyond the pure geometrical reconstruction of the surfaces. Indeed, being able to perform operations either on spatial and spectral bases leverage the analysis to the physical properties of the objects in a scene. Since the spectral response of the electromagnetic waves reflected by an object depends of its material, we can take advantage of the signal acquired at different bands to discriminate between different materials. This has huge potential in fields like food security, defense technologies, earth sciences and many more.

We refer the reader to [153] for a complete discussion of multi-spectral imaging techniques and applications.

2

Related Work

2.1 The Pinhole Camera Model

The *Pinhole Camera Model* assumes that all light rays entering the camera are converged by the lenses to a single point \mathbf{o} , called *Optical Center*, before hitting the sensor image plane Π which is parallel to the camera xy plane (Fig.2.1 ,Left). All the points in Π are described by the image coordinate system defined by the versors \vec{u}, \vec{v} . Conventionally, the optical center is placed behind the image plane assuming that light rays can pass through it to form the final image (Fig.2.1, Right). Mathematically, this is equivalent to the original case with the difference that, with the optical center behind, the obtained image is not vertical mirrored¹.

The camera z -axis, orthogonal to Π , is called the *optical axis*. Moreover, the distance (usually expressed in pixels) between the optical center and Π is the camera *focal length* whereas the projection of \mathbf{o} to the image plane (i.e. the intersection between the optical axis and Π) is a 2D point $\mathbf{c} = (c_x, c_y)$ called *principal point*.

This model implements a pure *central projection* in which the image $\mathbf{m} = (u, v)^T$ of any 3D point $\mathbf{M} = (X, Y, Z)^T$ is formed by the intersection of the straight line between \mathbf{o} and \mathbf{M} and the image plane Π . Consequently, \mathbf{o} , \mathbf{m} and \mathbf{M} are all collinear [65]. In Figure 2.2 the central projection process is illustrated. To describe this projection mathematically, we consider the point $\hat{M} = (X, Y, Z, 1)^T \in \mathbb{R}^4$ in *homogeneous coordinates* obtained by adding 1 as the last component of \mathbf{M} . Then, the equation relating \mathbf{M} to its projection \mathbf{m} is given by Equation 2.1.

$$\lambda \hat{m} = \lambda \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \hat{m}_s = \mathbf{P}\hat{M} = \mathbf{K}(\mathbf{R} \quad \vec{T})\hat{M} \quad (2.1)$$

In the equation, λ is any scale factor, $(\mathbf{R} \quad \vec{T})$ is the 3×4 matrix of camera extrinsic parameters and \mathbf{K} is the 3×3 matrix of the intrinsic parameters (f_x, f_y, s, c_x, c_y) disposed as in Equation 2.2.

¹We also assume the optical center behind the image plane throughout this thesis

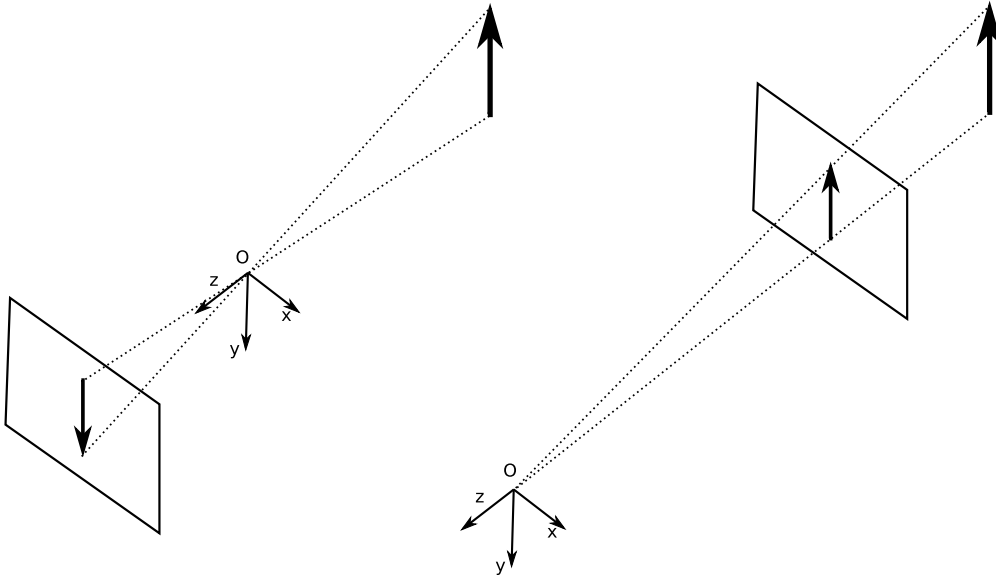


Figure 2.1: Left: Path followed by light rays entering the camera in a pinhole model. Right: The pinhole model conventionally used with the optical center placed behind the image plane. Note how the projection is substantially the same but is not vertical-mirrored.

$$\mathbf{K} = \begin{pmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix} \quad (2.2)$$

Here, c_x and c_y are the coordinates of the principal point described before. To allow non square pixels, the focal length f is divided into two distinct components, namely f_x and f_y . In most cases, it is safe to assume squared pixels so that $f_x = f_y = f$ holds. Finally, s is a parameter that controls the skewness of the two image axis. Extrinsic and intrinsic matrices all together form the *Projection Matrix* \mathbf{P} that contains all the necessary informations to transform scene points to the image plane.

Expressing points in homogeneous coordinates allows the description of the non-linear nature of the central projection as a linear operation (See Sec. 2.1.3). However, since the output of the projection would depend to a scale factor λ , a subsequent division for the last component of \hat{m}_s is required. As discussed before, $(\mathbf{R} \quad \vec{T})$ is subject to 6 degrees of freedom whereas \mathbf{K} is subject to 5. Consequently, the estimation of 11 independent parameters is required to describe the imaging process.

Sometimes is useful to reason in terms of an idealized coordinate system transcending the specific characteristics of a camera. Specifically, in motion and stereo applications it is common to operate on a *Normalized Coordinate System* with unitary focal length, principal point being at the origin of the image plane and zero skewness.

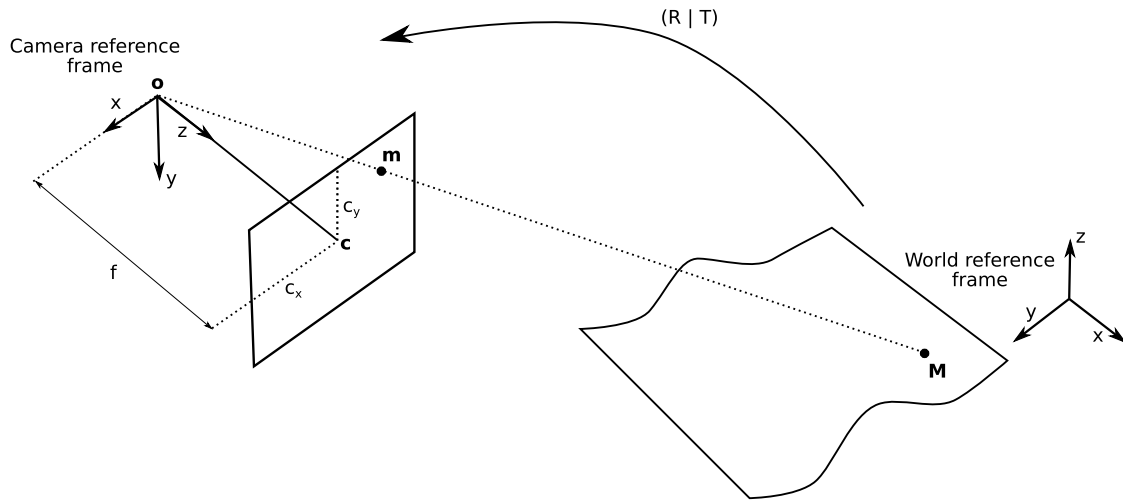


Figure 2.2: The projection process of a pinhole camera. First, a 3D point \mathbf{M} is transformed through the rigid motion \mathbf{R}, \vec{T} . Then, the transformed point is projected so that \mathbf{m} is the intersection between the line connecting \mathbf{o} and \mathbf{M} and the image plane.

A straightforward change in coordinate system can transform a camera image plane so that its projection matrix \mathbf{P}_n can be written as in Equation 2.3.

$$\mathbf{P}_n = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} (\mathbf{R} \quad \vec{T}) = \mathbf{HP} = \begin{pmatrix} \frac{1}{f} & 0 & -\frac{c_x}{f} \\ 0 & \frac{1}{f} & -\frac{c_y}{f} \\ 0 & 0 & 1 \end{pmatrix} \mathbf{P} \quad (2.3)$$

2.1.1 Lens Distortion

The pinhole camera model is by far the most widely used image formation model. The reasons behind its huge success among researchers and practitioners is due to several factors. First, its simplicity. In fact, the model is fully determined by its optical center and principal point. Given this basic information, the whole imaging process can be modelled as a direct projection of the scene to the image plane. Second, the availability of mathematical tools. Its plain formulation allows to easily apply a wide spectrum of powerful and well understood mathematical tools, ranging from epipolar geometry to projective invariance of conics and straight lines. Finally, its wide range of applicability.

However, due to the optical characteristics of camera lenses, the pure central projection of the pinhole camera model may not be sufficient to describe the image formation process. Indeed, especially for low-end hardware or wide-angle cameras a form of distortion can be observed that displace the points projected on the image plane.

If we include lens distortion, the 3D points transformation onto the image plane can be divided into 4 subsequent steps [179]:

1. A rigid motion to transform points from the world to the camera coordinate system. As seen in Sec. 2.1, this corresponds to a multiplication by a roto-translation matrix $(\mathbf{R} \quad \vec{T})$
2. A *perspective projection* that maps a 3D point M in camera reference plane to a 2D point $\mathbf{q} = (x, y)$ lying on an ideal image plane centered with the optical axis and placed at distance f from \mathbf{o} . Such operation is performed by the Equation 2.4

$$x = f \frac{X}{Z}, y = f \frac{Y}{Z} \quad (2.4)$$

3. A displacement of \mathbf{q} into $\mathbf{q}_d = (x_d, y_d)$ to model the lens distortion. Here, the central projected coordinates of \mathbf{q} are transformed to match what is obtained by the camera real projection. This operation can be mathematically described as in Equation 2.5

$$x_d = x + \delta_x, y_d = y + \delta_y \quad (2.5)$$

4. A final affine transformation to map 2D points from the ideal image plane to the sensor image plane. This transformation involves a translation with vector (c_x, c_y) and an optional scale that maps the measure of f to pixels.

In this setting, the lens distortion described by (2.5) is essentially causing a displacement of points in the idealized plane. Note that, in literature, the distortion can be defined in terms of a function that maps undistorted coordinates into distorted counterpart, or vice-versa. The latter (2.6) is mostly used because it directly describes the operation to be performed to undistort an image once taken from the camera. Moreover, the displacement δ_x, δ_y may be expressed as a function for which may not exist a closed-form inverse.

$$x = x_d + \delta_x, y = y_d + \delta_y \quad (2.6)$$

In the naive case with no distortion, both δ_x and δ_y will be 0 as in the seminal model described by Toscani [178]. Albeit simpler than previous proposals [40, 41], the model introduced by Tsai [179] was particularly successful for two reasons. First, is able to offer a good approximation of the imaging process (at least for moderately distorted cameras) and, second, an easy calibration procedure was available since its introduction. In Tsai's model, only the radial distortion [165] is taken into account so that image points are radially translated away from the center (Fig. 2.3).

Let $r = \sqrt{x_d^2 + y_d^2}$ being the radial distance of a point. Tsai's model approximates the distortion as a polynomial with two non-zero coefficients respectively for the second and fourth degree terms:

$$\delta_x = x_d(k_1 r^2 + k_2 r^4), \quad \delta_y = y_d(k_1 r^2 + k_2 r^4) \quad (2.7)$$

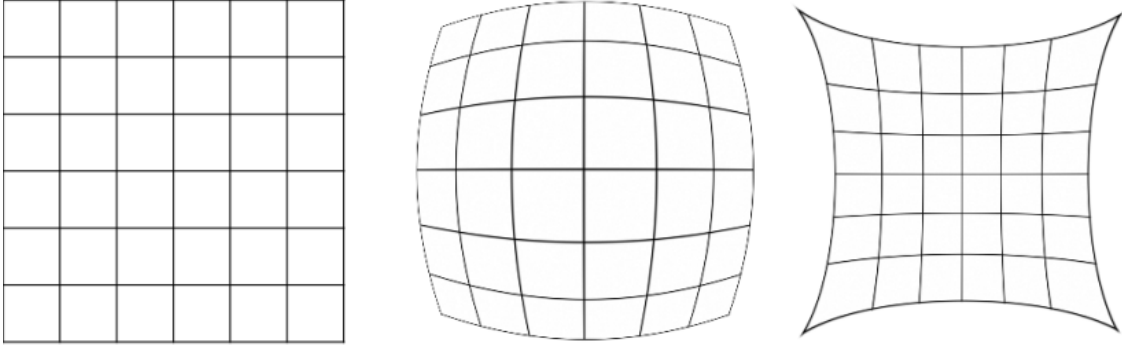


Figure 2.3: Different types of radial distortions. Left: Original undistorted image. Center: positive (barrel) distortion. Right: negative (pincushion) distortion.

It has been demonstrated in literature [42, 179] that radial distortion is mainly caused by imperfection in the radial curvature of the lens and, for most applications, can be estimated considering the approximation of the first term of r power series. Zhang, when introducing his well-known calibration procedure [206], adopted the same model with slight modifications.

Nevertheless, Weng [191] proposed a model for which δ_x and δ_y are given by a combination of radial (such in Tsai), tangential and thin prism distortion (2.8) caused by faulty lens positioning.

$$\delta_x = \delta_{xr} + \delta_{xt} + \delta_{xp}, \quad \delta_y = \delta_{yr} + \delta_{yt} + \delta_{yp} \quad (2.8)$$

In such setting, the tangential distortion that corrects for mis-alignments between the lenses along the optical axis is described by the two coefficients p_1 and p_2 in Equation 2.9².

$$\begin{aligned} \delta_{xt} &= p_1(3x^2 + y^2) + 2p_2xy \\ \delta_{yt} &= 2p_1xy + p_2(x^2 + 3y^2) \end{aligned} \quad (2.9)$$

Finally, the thin-prism distortion is modelled by the parameters s_1 and s_2 .

$$\delta_{xp} = s_1r^2, \quad \delta_{yp} = s_2r^2 \quad (2.10)$$

More recently, Claus and Fitgibbon [55] proposed a rational function as a replacement for the original polynomial term. This latter approach is currently one of the most successful, probably because of its inclusion in the OpenCV library [36]. Finally, Other recent approaches include variations on the number and type of parameters [124, 188], the enforcement of projective invariants for parameter estimation [61, 23] and extensions designed to work with highly distorted cameras [164, 171].

²Note that this model follows the convention described by equation (2.5)

2.1.2 Calibration techniques

Once a camera model is chosen, we are left to the problem to define and evaluate a calibration procedure apt to estimate the parameters governing its behaviour. Depending of the complexity of the model and the number of parameters, the task may not be easy nor numerically stable. In this section we concentrate on different techniques to calibrate a pinhole camera with various types of lens distortions described in Sec. 2.1.1. Other types of camera models will be discussed in Sec. 2.3.

Almost any calibration technique works by analyzing the 2D projection of a known physical *Calibration Target* once observed by the camera in (possibly) many different poses. Thereafter, by comparing the observed projection with the known target, the aim is to estimate the best possible set of parameters that minimize the difference between the expected and the observed projection.

In this terms, it is quickly clear that part of the process is the definition of a proper target. To begin with, we can give the following taxonomy on the calibration targets (and hence the related calibration techniques) which have been proposed over the years [133]:

3D Calibration targets: composed by a set of points (not collinear nor coplanar) for which the precise spatial position is known. For instance, a set of chessboard corners lying onto two or three coplanar planes [65] or a single plane moved back and forth with a known translation [179]. The accuracy obtainable with these targets is very high at a price of a difficult manufacturing process.

2D Calibration targets: composed by pattern of points lying on a planar surface [168]. Targets falling to this category are probably the most used and studied for their simplicity to use. Indeed, the manufacturing can be performed with a consumer ink-jet printer and, differently from [179], no complicated machinery to control the motion of the target is needed.

1D Calibration targets: composed by a sets of collinear features possibly moving in a line around a point [207]. Such targets are the weapon of choice to calibrate complex camera networks for which the simultaneous observation of a planar target by all the cameras is not feasible.

Finally, it worth to be noted that camera calibration can be performed with no target at all by observing a static scene from different point of views. The process is based on the ability to recognize same (unknown) 3D points on a scene from the relative projections into the sequence of different poses [131].

The vast majority of calibration methods [191, 206, 178, 157] assume a known set of 3D-2D points correspondences and start by an initial linear estimation of the projection matrix \mathbf{P} . Thereafter, \mathbf{R} , \vec{T} and \mathbf{K} are recovered by factorizing \mathbf{P} and a non-linear refinement optimization is performed to minimize the geometric error between the observed 2D points and the projection of the 3D target points using the estimated calibration.

Since the target is known, the 3D position of each feature in world reference frame can be related to its 2D projection on the image. Many different targets have been proposed in literature mostly composed by corner or circular based features [184]. For example, in the camera calibration chapter of [133], a checker pattern is used. Conversely, in [50] a pattern composed by circular control points is used together with a bias correction method to improve the accuracy. Finally, in [19] a bundle adjustment step is performed to reduce the manufacturing error of a chessboard target. In Chapters 3 and 4 some novel circular features based targets (originally designed for camera pose estimation) are proposed and evaluated.

After collecting such correspondences, almost all methods starts by estimating the matrix \mathbf{P} . Indeed, the direct recovery of \mathbf{R} , \vec{T} and \mathbf{K} is quite non-linear hence causing the problem being very difficult to optimize whereas the recovery of whole coefficients of \mathbf{P} is linear. These methods start by linearize \mathbf{P} into the vector $\vec{p} = (P_{1,1}, P_{1,2}, \dots, P_{3,4})^T$ where $P_{i,j}$ is the element of \mathbf{P} at row i and column j . Then, for each 3D point $M_i = (X_i, Y_i, Z_i)^T$ and its corresponding 2D point $m_i = (u_i, v_i)^T$, Equation 2.1 can be rewritten as in (2.11).

$$\begin{aligned} u_i &= \frac{P_{1,1}X_i + P_{1,2}Y_i + P_{1,3}Z_i + P_{1,4}}{P_{3,1}X_i + P_{3,2}Y_i + P_{3,3}Z_i + P_{3,4}} \\ v_i &= \frac{P_{2,1}X_i + P_{2,2}Y_i + P_{2,3}Z_i + P_{2,4}}{P_{3,1}X_i + P_{3,2}Y_i + P_{3,3}Z_i + P_{3,4}} \end{aligned} \quad (2.11)$$

Whence, by re-arranging the factors with respect to \vec{p} , we obtain the formulation in (2.12) which gives 2 equations in 12 unknowns:

$$\mathbf{D}_i \vec{p} = \begin{pmatrix} X_i & Y_i & Z_i & 1 & 0 & 0 & 0 & 0 & u_i X_i & u_i Y_i & u_i Z_i & u_i \\ 0 & 0 & 0 & 0 & X_i & Y_i & Z_i & 1 & v_i X_i & v_i Y_i & v_i Z_i & v_i \end{pmatrix} \vec{p} = \vec{0} \quad (2.12)$$

By considering n different correspondences, a matrix \mathbf{A} is created by stacking $\mathbf{D}_1 \dots \mathbf{D}_n$ by rows obtaining a linear system of $2n$ equations in 12 unknowns. Since \mathbf{P} is defined up to scale, an additional normalization constraint has to be applied to the solution to avoid the trivial case $\vec{p} = \vec{0}$. Usually, unitary $\|\vec{p}\| = 1$ is imposed resulting the optimization problem in (2.13) whose optimal is given by the singular vector associated with the smallest singular value of \mathbf{A} [133].

$$\min_{\vec{p}} \|\mathbf{A}\vec{p}\|^2 \quad \text{subject to } \|\vec{p}\| = 1 \quad (2.13)$$

A similar procedure, but using homographic constraints, is given by Zhang [206] when dealing with 2D planar targets. If all the target points are not coplanar, the estimated matrix \mathbf{P} can be factorized into extrinsic and intrinsic matrices by considering that $\mathbf{P} = \mathbf{K}(\mathbf{R} \quad \vec{T})$ and dividing \mathbf{P} into the sub-matrix \mathbf{B} and the vector \vec{b} composed respectively by the first three columns and the last column of \mathbf{P} .

$$\begin{aligned}\mathbf{B} &= \mathbf{KR} \\ \vec{b} &= \mathbf{A}\vec{T}\end{aligned}\tag{2.14}$$

Consequently, from (2.14), the matrix \mathbf{K} can be obtained as $\mathbf{K} = \mathbf{BB}^T$ with a subsequent normalization so that the last element $K_{3,3} = 1$. At this point, the extrinsics are obtained as:

$$\begin{aligned}\mathbf{R} &= \mathbf{K}^{-1}\mathbf{B} \\ \vec{T} &= \mathbf{K}^{-1}\vec{b}\end{aligned}\tag{2.15}$$

This estimation of camera parameters minimize the algebraic error obtained from point correspondences. However, to obtain a physically meaningful geometric error between all the pairs, a non-linear optimization step is performed via Levenberg-Marquardt [138] that can also include a non-linear distortion function discussed in Section 2.1.1.

2.1.3 Projective planes, conics and ellipse fitting

The central projection assumed by the pinhole camera model allows the description of the image formation process by means of the mathematical framework of *projective geometry* [85]. Projective geometry can be defined in any numbers of dimensions but, for the sake of this introductory part, we will restrict to the two dimensional projective plane \mathbb{P}^2 related to the two dimensional euclidean space. In this sense, when we observe a geometric primitive lying on a two-dimensional subspace of a three dimensional scene, its projection to the image plane can be described as a projective transformation of planes from which we can derive some interesting properties.

Points lying on an euclidean plane can be described as vectors $p = (x, y)^T \in \mathbb{R}^2$. As briefly introduced in previous sections, to represent points lying on a projective plane we take advantage of *homogeneous coordinates* that allow the definition of projectivities (i.e. invertible mappings between \mathbb{P}^2 to itself) in term of non-singular 3×3 matrices. In homogeneous coordinates, we represent p as all vectors in \mathbb{R}^3 in the form $p^h = (kx, ky, k)^T$ for any $k \neq 0$. Consequently, points in euclidean space are represented by all the equivalence classes of three dimensional vectors where two elements are in the same class if they differ by a non zero scale factor. It is easy to go back and forth from \mathbb{P}^2 to \mathbb{R}^2 with simple operations. Specifically, to describe p in \mathbb{P}^2 we just add a unitary third component (i.e. $p_h = (x, y, 1)^T$). Conversely, a point in homogeneous coordinates can be transformed in euclidean 2D space by considering only the first two components divided by the third (i.e. if $p_h = (X, Y, W)^T$, $W \neq 0$, then $p = (X/W, Y/W)^T$).

A line in \mathbb{R}^2 is represented by the locus of points $p = (x, y)$ so that $ax + by + c = 0$ holds. In projective space \mathbb{P}^2 , the same line can be described as $(X, Y, W)(a, b, c)^T = 0$

where $\vec{u} = (a, b, c)^T$ represent the line and $\vec{x} = (X, Y, Z)^T$ represent each point on a line. If we observe \vec{u} , we realize that in \mathbb{P}^2 points and lines are described in the same way, which leads to a simple expression to find the intersection x of two lines \vec{u} and \vec{u}' by means of vector cross product:

$$x = \vec{u} \times \vec{u}' \quad (2.16)$$

Furthermore, if we try to compute the intersection between two parallel lines $\vec{u} = (a, b, c)$ and $\vec{u}' = (a, b, c')$ using (2.16) we obtain (discarding the scale factor $c - c'$) $x = (b, -a, 0)$. Clearly, this point cannot be transformed back to \mathbb{R}^2 since the last coordinate is zero. This agrees to the fact that two parallel lines have no intersection point in the euclidean space. However, in projective geometry we have a concise yet powerful way to reason about normal points and points lying at the infinite, called *ideal points*. Finally, all ideal points line on a *ideal line* represented by the vector $(0, 0, 1)^T$ which, once again, is threatened in the same way of all other non-ideal lines.

A planar projective transformation is a linear transformation that can be represented by any non-singular 3x3 matrix \mathbf{H} :

$$\begin{pmatrix} X' \\ Y' \\ W' \end{pmatrix} = \mathbf{H}x = \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{pmatrix} \begin{pmatrix} X \\ Y \\ W \end{pmatrix} \quad (2.17)$$

Consequently, any projective transformation transform lines in lines and preserve the incidence.

Similarly as we did before, we can now focus our attention on conics that can be represented by second degree polynomials in the form $ax^2 + bxy + cy^2 + dx + ey + f = 0$. In homogeneous matrix form the conic ξ is represented by a matrix:

$$\mathbf{Q} = \begin{pmatrix} a & b & d \\ b & c & f \\ d & f & g \end{pmatrix}$$

such that, for each point $\mathbf{x} = (x, y, 1)^T \in \xi$, it holds that $\mathbf{x}^T \mathbf{Q} \mathbf{x} = 0$. Different values of the parameters will lead to different type of conics (i.e. circles, ellipses or hyperbolas). Again, any projectivity transforming the projective space \mathbb{P}^2 in which the conic lie will result in (possible different) type of conic. For this reason, ellipses present in a scene (and circles, as special case) are interesting geometric primitives since they remain ellipses after being projected into the image plane. Since \mathbf{Q} is defined up to any non-zero scale, it is subject to 5 degrees of freedom and, consequently, can be estimated by at least 5 points lying on that conic.

Different approaches have been proposed in literature to provide a robust and precise estimation of ellipse parameters from its projection on the image plane. The seminal work by Bookstein [35] implementing a least square fitting of ellipse parameters has been extended by Cabrera and Meer [46] my means of a general method for eliminating the bias of nonlinear estimators. Furthermore, Fitzgibbon et. al. [72] incorporate

the ellipticity to normalization so that is ellipse-specific and far more robust and computationally inexpensive than estimating a general conic. More recently, it is worth to be noted the maximum likelihood method by Kanatani [100] and Ouellet's estimator defined as linear operator [146].

2.2 Camera pose estimation

Sometimes it might be the case that the intrinsic parameters of the camera are known whereas the rigid motion that localizes the device with respect to the world reference frame is not. This is usually the case in which the camera internal geometry is not changing, for example if using fixed lenses, but its motion with respect to the scene must be recovered to localize the observer.

The camera rigid motion recovery techniques can be divided into two classes, namely: *Absolute Orientation* (AO) and *Perspective-n-Point* (PnP) [21]. The first class assumes a set of 3D points expressed in a coordinate frame Φ_1 and a corresponding set of 3D points expressed in a coordinate frame Φ_2 . The goal is to recover the rigid motion that registers all the points in Φ_1 with the points in Φ_2 assuming a noise term ϵ . Formally, we seek for a rotation matrix \mathbf{R} and a translation vector \vec{T} such that the squared distance between the points in Φ_1 ($p_1 \dots p_n$) and the corresponding points in Φ_2 ($q_1 \dots q_n$), transformed through $(\mathbf{R} \quad \vec{T})$, is minimum:

$$\operatorname{argmin}_{\mathbf{R}, \vec{T}} \sum_i^N \|p_i - (\mathbf{R}q_i + \vec{T})\|^2 \quad (2.18)$$

In this class we mention the method by Horn et. al. [92], Horn [90], Arun et. al [24] and Walker et. al. [186]. On the other hand, in the Perspective-n-Point pose estimation we assume to know a set of 3D points ($p_1 \dots p_n$) in Φ_1 reference frame and their relative 2D projections ($u_1 \dots u_n$) onto the camera image plane in a calibrated (i.e. intrinsic parameters are known) environment. Most of PnP approaches differentiate on the optimization type (i.e. linear vs. non-linear) and on the number of points used. A linear solution to the problem with $n = 3$ was the first studied [22] but can find optimal solutions only up to 4 different configurations. Conversely, if more than 3 points are taken into account, there exist linear methods that are guaranteed to find a unique solution [150, 152]. Furthermore, linear models with 5 or even more than 6 points have been proposed [86] but are rarely used because minor improvements of pose accuracy come at the cost to find many correct point-to-point correspondences that may become an expensive task in case of outliers.

Finally, non-linear iterative solutions to PnP problem have been proposed in [115, 123]. For an evaluation of some of the pose estimation algorithms we refer the reader to [21].

2.2.1 Fiducial Markers

Similar to the complete camera calibration case, pose estimation methods rely on a set of 3D-2D correspondences that must be provided. This implies the ability to identify objects present in a scene or, at least, to recognize similar features from different point of views. The most reliable way to provide such correspondences is to place artificial objects in a scene with known geometry so that the identification of the composing points is fast, simple and reliable.

In this contest, we call *fiducial marker* any artificial object consistent with a known model that is placed into a scene in order to supply a reference frame. Currently, such artefacts are unavoidable whenever a high level of precision and repeatability in image-based measurement is required, as in the case of vision-driven dimensional assessment task such as robot navigation and SLAM [187, 60, 66], motion capture [202, 47], pose estimation [203, 199], camera calibration [62, 99] and of course in field of augmented reality [205, 107].

While in some scenarios approaches based on naturally occurring features have been shown to yield satisfactory results, they still suffer from shortcomings that severely limit their usability in uncontrolled environments. Specifically, the lack of a well known model limits their use in pose estimation. In fact, while using techniques like bundle adjustment can recover part of the pose, the estimation can be only up to an unknown scale parameter; further, the accuracy of the estimation heavily depends on the correctness of localization and matching steps.

Moreover, the availability and distinctiveness of natural features is not guaranteed at all. Indeed the smooth surfaces found in most man-made objects can easily lead to scenes that are very poor in features.

Finally, photometric inconsistencies due to reflective or translucent materials severely affect the repeatability of the point descriptors, jeopardizing the correct matching of the detected points. For this reasons, it is not surprising that artificial fiducial tags continue to be widely used and are still an active research topic.

Markers are generally designed to be easily detected and recognized in images produced by a pinhole camera. In this sense they make heavy use of the projective invariance properties of geometrical entities such as lines, planes and conics.

One of the earliest invariance used is probably the closure of the class of ellipses to projective transformations. This implies that ellipses (and thus circles) in any pose in the 3D world appear as ellipses in the image plane. This allows both for an easy detection and a quite straightforward rectification of the plane containing any circle.

With their seminal work, Gatrell et al. [77] propose to use a set of highly contrasted concentric circles and validate a candidate marker by analyzing the compatibility between the centroids of the detected ellipses. By alternating white and black circles a few bits of information can be encoded in the marker itself. In the work proposed in [54] the concentric circle approach is enhanced by adding colors and multiple scales. Later, in [109] and [140], dedicated “data rings” are added to the marker design.

A set of four circles located at the corner of a square is adopted in [56]: in this case

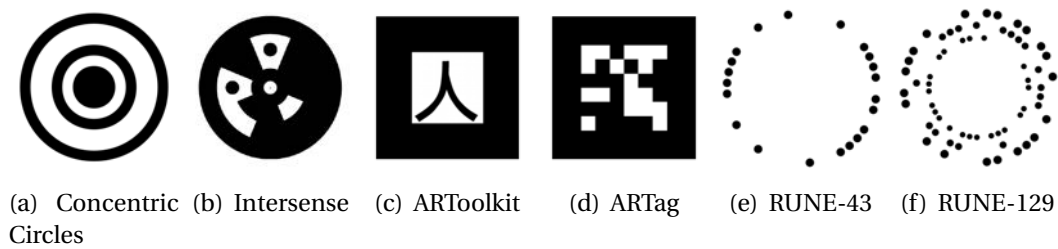


Figure 2.4: Some examples of fiducial markers that differ both for the detection technique and for the pattern used for recognition. In the first two, detection happens by finding ellipses and the coding is respectively held by the color of the rings in (a) and by the appearance of the sectors in (b). The black square border enables detection in (c) and (d), but while ARToolkit uses image correlation to differentiate markers, ARTag relies in error-correcting binary codes. Finally, in (e) and (f) we show two classes of RUNE-Tags fiducial markers described in Chapter 4.

an identification pattern is placed in the middle of the four dots in order to distinguish between different targets. This ability to recognize all the viewed markers is really important for complex scenes where more than a single fiducial is required; furthermore, the availability of a coding scheme allows for an additional validation step and thus lowers the number of false positives.

Circular features are also adopted in [181], where a set of randomly placed dots are used to define distinguishable markers that can be detected and recognized without the need for a frame. In this case, to attain robustness and to avoid wrong classification a large number of dots is required for each marker, thus leading to a likely high number of RANSAC iterations.

Collinearity, that is the property of points that lie on a straight line of remaining aligned after any projective transformation, is another frequently used invariant. Almost invariably this property is exploited by detecting the border edges of a highly contrasted quadrilateral block. This happens, for instance, with the very well known ARToolkit [103] system which is freely available and has been adopted in countless virtual reality applications. Thanks to the ease of detection and the high accuracy provided in pose recovery [126], this solution is adopted also in many recent marker systems, such as ARTag [67] and ARToolkitPlus [185]. The latter two methods replace the recognition technique of ARToolkit, which is based on image correlation, with a binary coded pattern.

Finally, many papers suggest the use of the cross-ratio among detected points [175, 180, 117, 122], or lines [183] as invariant properties around which to build marker systems. A clear advantage of the cross-ratio is that, being a projective invariant, the recognition can be made without the need of any rectification of the image. Unfortunately, the ease of detection offered by the use of the cross-ratio often comes at the price of a high sensitivity to occlusions or misdetection. In fact, spurious or miss-



Figure 2.5: Example of a fish eye lens (Left) with a sample picture of the result obtainable (Right). Note the severe distortion that cause straight lines to appear curved.

ing detection completely destroy the invariant structure. Further, cross-ratios exhibit a strongly non-uniform distribution [98], which in several situation limits the overall number of distinctively recognizable patterns.

2.3 Non-Pinhole Models

As discussed in section 2.1, the simplest formalization of the imaging process is the pinhole camera. This basic model can be calibrated by solving a linear system that relates the coordinates of reference point in the scene with their projections on the image plane. Nevertheless, the range of possible imaging capabilities of a pure pinhole model are quite limited since it is quickly clear that increasing the field of view requires a reduction of the focal length that is bounded by geometric factors. For this reason, camera lenses introduce different types of radial distortion as a trade-off between the field of view and the central projection constraint. Regardless its level of sophistication, any pinhole-based model is geometrically unable to properly describe cameras with a frustum angle that is near or above 180 degrees.

In addition to pinhole models with lens distortion, in the recent past many non-pinhole imaging devices have been proposed (See Sec. 2.3.2). These special types of cameras can provide a projection from different points of view of the same scene. This can be exploited to implement single lens stereoscopy [15] or digital refocusing [70].

2.3.1 Wide-angle, Fish eye and Catadioptric Cameras

Wide-angle and Fish eye lenses [136] are widely used in many application for which a wide field of view is needed (Fig. 2.5). The projection model involved introduces severe distortion so that straight lines appear curved into the image plane. To overcome this,

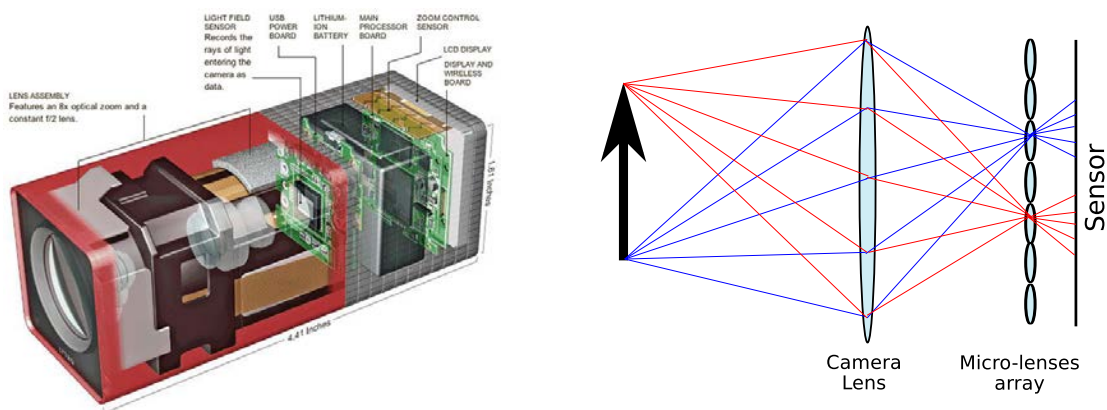


Figure 2.6: Left: A schematic representation of the Lytro™ camera. Right: The scene projection process on a plenoptic camera based on array of micro lenses. A standard camera lens focuses the scene onto the micro lenses array that act like multiple small pinhole cameras converging the captured rays to the device sensor. This way, the whole light field entering the main lens can be acquired.

several different parametric models have been proposed. Some of them try to modify the captured image in order to follow the original pinhole behaviour [124] while others introduce totally new image formation processes. For example, Kannala and Brandt [101] propose a model in which the distance between a point in the image plane and the principal point is a polynomial function of the angle θ between the principal axis and the incoming ray.

An interesting classes of devices using a combination of lenses and mirrors are called *Catadioptric*. Usually, the size and geometry of the mirror is carefully studied so that the whole device exhibit a single point of view but may present a multitude of combined distortion (either pincushion or barrel) into the same image. A review of different catadioptric systems is given by Nayar and Baker [142].

Regardless the model used to describe these devices, the calibration process is usually tedious and affected to highly non-linear constraints.

2.3.2 Plenoptic Cameras

If we drop the single point of view requirement we obtain a whole new class of imaging devices called *Plenoptic* that can be either composed by a network of independent pinhole cameras [172] or micro-lens arrays such the nowadays famous Lytro™[78] (Fig. 2.6).

The fundamental concept behind such kind of devices is the *plenoptic function* [16] that describe the complete dense set of light rays travelling into a region of space at a specific time. More formally, the plenoptic function describes the light intensity of

the ray passing through a point \mathbf{p} , travelling along the direction \mathbf{d} at a time t for a specific wavelength λ . It's clear that, if an estimation of such function inside a volume is known, a complete three-dimensional holographic reconstruction of a dynamic scene would be possible. Practically speaking, a sparse estimation of this function is usually recovered by considering a set of micro pinhole cameras (i.e. composed by only few measured exiting rays) overlapping each other.

2.3.3 Raxel-based Unconstrained Imaging Models

Given this proliferation of different models, the desire for an unifying approach is quite natural. The most general imaging model, that associates an independent 3D ray to each pixel, would in principle be able to describe any kind of imaging system, regardless of the optical path that drives each ray to the sensitive elements of the device. However, the complete independence of millions of rays makes its calibration a daunting task as each of them needs several 2D to 3D correspondences to be properly constrained.

This problem was first addressed in [80], where unconstrained rays (here called *raxels*) are calibrated exploiting their intersections with a target (an encoded laptop monitor) that moves along a translating stage. Such intersections are identified by means of Gray coding and are evaluated as the average over several different shots. This approach is somewhat limited by the fact that the pose of the calibration planes must be known (i.e. the method depends on the accuracy of the translating stage), and the paper does not assess the accuracy of the obtained calibration.

A method for the calibration of the general model and unknown poses is proposed in [170]. The authors discuss both the case of non-central and perspective camera, however, in the latter case, the parametrization process has proven to be rather complicated when using planar calibration objects [169]. Further, the paper does not describe a specific systematic setup for gathering 2D to 3D correspondences for all the camera rays and the experimental evaluation is performed qualitatively in a subset of the imaging sensor. The practical usage of multiple grids for calibrating generic non-perspective device is investigated in [151]. Again, the method is designed and well-suited for catadioptric, spherical, multiview and other types of non-central cameras.

In Chapter 6 we propose the use of an unconstrained model even in standard central camera settings dominated by the pinhole model, and introduce a novel calibration approach that can deal effectively with the huge number of free parameters associated with it, resulting in a higher precision calibration than what is possible with the standard pinhole model with correction for radial distortion. This effectively extends the use of general models to settings that traditionally have been ruled by parametric approaches out of practical considerations.



Figure 2.7: An example of optical flow estimation from two images (Left and Center) taken from a moving camera. At each point of the first image is associated a displacement (flow) vector to the corresponding point on the second.

2.4 Optical Flow Estimation

Optical Flow estimation is historically one of the most studied problems in computer vision. Following Horn’s definition [89], the *Optical Flow* is “*the apparent motion of brightness patterns in the image*”. Thus, given two projections of a same scene, its estimation is essentially the problem of finding a dense correspondence between all the pixels of the first image to the second (i.e. the apparent pixel motion from the first to the second image). For instance, in Fig. 2.7 an example of optical flow is shown from two images taken at different camera poses.

Typical applications can be found in image segmentation [189], multiple-object detection and tracking [159], visual odometry for robots [48] and video compression [139].

Most of the approaches to optical flow estimation consider the problem in terms of a global energy minimization of a functional defined as a combination of a data and a regularization (or prior) term:

$$E_{\text{global}} = E_{\text{data}} + \lambda E_{\text{reg}} \quad (2.19)$$

The E_{data} term measures the photometric consistency of the corresponding pixels mapped by the flow whereas E_{reg} favors smoothly varying flow fields. Most of the time, the data term is designed by assuming that the brightness intensity of a pixel is maintained across the two images (*brightness constancy assumption*). If we denote with $I(x, y)$ the intensity of a pixel at coordinates x, y in the image I and the flow as a function $f : \mathbb{R}^2 \rightarrow \mathbb{R}^2$, the brightness constancy can be written as in Equation 2.20 which is usually linearized considering the first order Taylor expansion yielding the approximation described by Equation 2.21.

$$I_1(x, y) = I_2(f(x, y)) \quad (2.20)$$

$$I_2(x, y) = I_1(x, y) + f(x, y)^T \begin{pmatrix} \frac{\partial I}{\partial x} \\ \frac{\partial I}{\partial y} \end{pmatrix} \quad (2.21)$$

The seminal formulation by Horn and Schunck [93] was based on this assumption. However, since the brightness intensity of pixels can vary for changes in illumination (i.e. shadows, specular highlights, etc.) it has been proposed to use different robust photometric features like image gradients [43] or SIFT descriptors [121].

In its simplest formulation, the regularization term E_{reg} is defined in term of the squared (L2) norm of the gradient of the function f :

$$E_{\text{reg}} = \int_x \int_y \|\nabla f(x, y)\|^2 dy dx \quad (2.22)$$

but recent proposed algorithms use a robust (L1) norm [43] or Total Variation [190]. The choice of a regularization function that is not penalized by occlusion boundary regions is of pivotal importance to estimate a high-quality optical flow and for which great efforts have been devoted in the past.

Finally, a huge selection of optimization strategies have been proposed over the years. Some authors propose to use a gradient descent algorithm which updates the flow function by taking incremental steps to the negative direction of the energy function gradient [26]. Since the regularization function is often described in terms of high order derivatives of f , many authors [209, 43, 196] treat the optimization as a calculus of variations. Specifically, they assume an energy functional that can be written as a function of the coordinates (x, y) , the flow $f(x, y) = (f_u(x, y) \ f_v(x, y))^T$ and the first order partial derivatives of the flow function:

$$E_{\text{global}} = \int_x \int_y E\left(x, y, f_u(x, y), f_v(x, y), \frac{\partial f_u}{\partial x}, \frac{\partial f_u}{\partial y}, \frac{\partial f_v}{\partial x}, \frac{\partial f_v}{\partial y}\right) dy dx \quad (2.23)$$

Such formulation can be transformed in terms of Euler-Lagrange equations (2.24) that can be solved as a system of partial differential equations.

$$\begin{aligned} \frac{\partial E_{\text{global}}}{\partial f_x} - \frac{\partial}{\partial x} \frac{\partial E_{\text{global}}}{\partial \frac{\partial f_u}{\partial x}} - \frac{\partial}{\partial y} \frac{\partial E_{\text{global}}}{\partial \frac{\partial f_u}{\partial y}} &= 0 \\ \frac{\partial E_{\text{global}}}{\partial f_y} - \frac{\partial}{\partial x} \frac{\partial E_{\text{global}}}{\partial \frac{\partial f_v}{\partial x}} - \frac{\partial}{\partial y} \frac{\partial E_{\text{global}}}{\partial \frac{\partial f_v}{\partial y}} &= 0 \end{aligned} \quad (2.24)$$

Furthermore, sparse-to-dense approaches can start from an initial sparse estimation of the flow that can easily handle large motions of scene objects. Then, the flow is diffused to all the image regions taking advantage of the smoothness prior [44, 196].

For a complete in-depth taxonomy of the most important optical flow methods, we refer the reader to [27].

2.5 Reflectance Modelling

In computer vision, the modelling and recovery of photometric parameters is a topic of pivotal importance for purposes of surface analysis and image understanding. In Section 2.1 we focused our attention on geometric aspects of image formation process, specifically, how points in 3D world reference frame get mapped into the image plane. However, when we start reasoning on the intensity and frequency of light radiation that is acquired by the camera, we enter in the field of *Radiometry* that describes energy transfers, in terms of electromagnetic waves, that occur from light sources, various surface areas and the imaging sensor.

Our analysis starts from a certain amount of incoming energy, called *Irradiance*, that is collected by the imaging sensor at a certain wavelength $I(\lambda)$. This energy is a result of the emission performed by an *Illuminant* $L(\lambda)$ (whose intensity is also wavelength dependent) and the fraction of incident light on a scene surface that gets reflected and scattered toward the image sensor. This scattering process depends on the foreshortening of each surface patch (i.e. the area of the surface viewed from a light source), the angle with respect to the observer (i.e. the camera) and, finally, by intrinsic properties of the surface such the material. A *Reflectance Model* describes the fraction of energy that is reflected from a surface that gets illuminated by a light source $L(\lambda)$, as a function of the illuminant, the local surface geometry (*shading*) and the object material (*reflectance*).

Since the estimation of illuminant and material reflectance are mutually interdependent, the problem of recovering physically meaningful parameters that govern the image formation process is closely related to the ability to resolve the intrinsic material reflectance from their trichromatic colour images captured under varying illumination conditions. Existing methods often rely upon the use of statistics of illuminant and material reflectance or draw upon the physics-based analysis of local shading and specularities of the objects in the scene.

Statistics-based approaches often employ Bayes's rule [37] to compute the best estimate from a posterior distribution by standard methods such as maximum a posteriori (MAP), minimum-mean-squared error (MMSE) or maximum local mass (MLM) estimation. The illuminant and surface reflectance spectra typically take the form of a finite linear model with a Gaussian basis [28, 69], where a correlation matrix is built for a set of known plausible illuminates to characterise all the possible image colours (chromaticities) that can be observed.

Contrary to these statistics-based approaches, physics-based colour constancy analyses the physical processes by which light interacts with the object surface [112, 111, 162].

Probably the simplest but effective model was proposed by Lambert [110]. Such model is based on two simple assumptions formally described in equation (2.25). First, the reflected energy (radiance) is proportional to the illuminant $L(\lambda)$ (depending only on the wavelength) and the material reflectance function $S(u, \lambda)$ (depending by the location u and the wavelength λ). Second, the amount of reflected energy is a function

of the cosine angle θ between the light position and the surface normal.

$$I(u, \lambda) = \frac{1}{\pi} L(\lambda) S(u, \lambda) \cos(\theta) \quad (2.25)$$

Lambertian model accounts only for pure diffusive surface reflection. To include also a specular component, the *dichromatic model* has been introduced by Shafer [163]. This model assumes a direction-independent diffuse component whereas the specular part is a function of the observer direction, as described by equation (2.26).

$$I(u, \lambda) = g(u) L(\lambda) S(u, \lambda) + k(u) L(\lambda) \quad (2.26)$$

The wavelength-dependent illuminant contributes to the final irradiance with two components. The former depends on the reflectance $S(u, \lambda)$ and is weighted by a shading factor $g(u)$ that varies within the surface. The latter is just the pure illuminant weighted by a specular factor $k(u)$ that also varies within the surface as it depends on the observer direction. For diffuse lambertian surfaces, the shading factor is usually a function of the cosine between the light and the surface normal whereas the specular factor is related to the *Fresnel Reflection Coefficients*.

Regarding specularities and shading, there have been several attempts to remove specular highlights from images of non-Lambertian objects. For instance, Brelstaff and Blake [38] used a thresholding strategy to identify specularities on moving curved objects. Conversely, Narasimhan *et al.* [141] have formulated a scene radiance model for the class of “separable” Bidirectional Reflectance Distribution Functions (BRDFs). More recently, Zickler *et al.* [208] introduced a method for transforming the original RGB colour space into an illuminant-dependent colour space to obtain photometric invariants.

Other alternatives elsewhere in the literature aiming at detecting and removing specularities either make use of additional hardware [143], impose constraints on the input images [119] or require colour segmentation [108].

I

Calibrating with circular features

3

A Projective Invariants Based Fiducial Marker Design

Visual marker systems have become an ubiquitous tool to supply a reference frame onto otherwise uncontrolled scenes. Throughout the last decades, a wide range of different approaches have emerged, each with different strengths and limitations. Some tags are optimized to reach a high accuracy in the recovered camera pose, others are based on designs that aim to maximizing the detection speed or minimizing the effect of occlusion on the detection process. Most of them, however, employ a two step procedure where an initial homography estimation is used to translate the marker from the image plane to an orthonormal world where it is validated and recognized.

In this chapter we present a general purpose fiducial marker system that performs both steps directly in image-space. Specifically, by exploiting projective invariants such as collinearity and cross-ratios, we introduce a detection and recognition algorithm that is fast, accurate and moderately robust to occlusion. Moreover, several real-world applications are proposed, ranging from camera calibration to projector-based augmented reality.

3.1 Introduction

In this chapter, we introduce a novel visual marker system that uses the cross-ratio and other projective invariants to perform both detection and recognition in the image plane, without requiring the estimation of an homography or any other technique of perspective correction. Further, our approach introduces some redundancy by replicating the same pattern on different sides, which can be exploited to obtain a moderated robustness to occlusion or to lower the false positive rate. In addition, the detection and recognition algorithms are both efficient and very simple to implement. In the experimental section we validate the proposed approach by comparing its performance with two widely used marker systems under a wide range of noise sources applied to synthetically generated scenes. Finally, we also tested the effectiveness of the novel marker when dealing with real images by using it to solve a number of different real-world measurement tasks and applications.

3.2 Image-Space Fiducial Markers

The proposed marker, which we named *Pi-Tag* (*Projective invariant Tag*), exhibits a very simple design. It is made up of 12 dots placed on the sides of a square: four dots per side, with the corner dots shared. There are two distinct configurations of the dots and each is repeated in two adjacent sides. See for example the marker in Figure 3.2(a): The top and left sides show the same configuration, and so do the bottom and right ones. The two different configurations are not random. In fact they are created in such a way that the cross-ratio of the two patterns is proportional via a fixed constant δ .

The interplay between the detection of these cross-ratios in the image plane and other invariants such as straight lines and conics projections allows for a simple and effective detection and recognition approach for the Pi-Tag.

3.2.1 Projective invariants

Our approach relies on four type of projective invariants. Namely, the invariance of the class of ellipses, collinearity, angular ordering (on planes facing the view direction) and cross-ratio.

The invariance of the class of ellipses has been extensively exploited in literature. Circular dots are easy to produce and, since they appear as ellipses under any projective transformation, they are also easy to detect by fitting on them a conic model with a low number of parameters. In addition, while the center of the detected ellipses is not preserved under perspective, if the original dots are small enough, the localization error has been shown to be negligible for most practical purposes [129].

Other advantages of the elliptical fitting include the ability of using the residual error to filter out false detections and to perform gradient-based refinements. For this

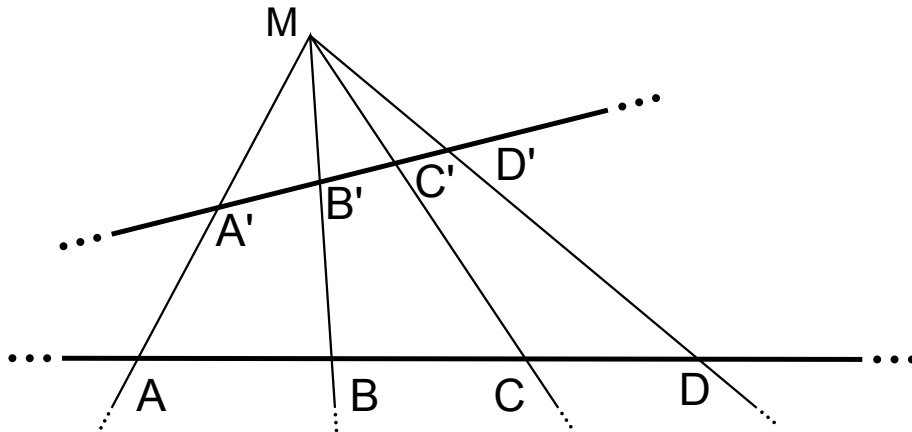


Figure 3.1: The cross-ratio of four collinear points is invariant to projective transformations. $cr(A, B, C, D) = cr(A', B', C', D')$

and other reasons, dots are widely adopted also for accurate tasks such as lens distortion correction, and stereo calibration.

Given a set of points, projective geometry preserves neither distances nor the ratios between them. Fortunately, there are some interesting properties that remain invariant and can be put to use. One is the angular ordering of coplanar points. That is, if we take three points defining a triangle, once we have established an ordering on them (either clockwise or anti-clockwise), such ordering is maintained under any projective transformations that looks down to the same side of the plane.

The second invariant is collinearity and derives from the fact that straight lines remain straight under perspective transformations. Almost all rectangular fiducial markers rely on this property in the detection stage by finding lines in a scene using a wide range of different techniques.

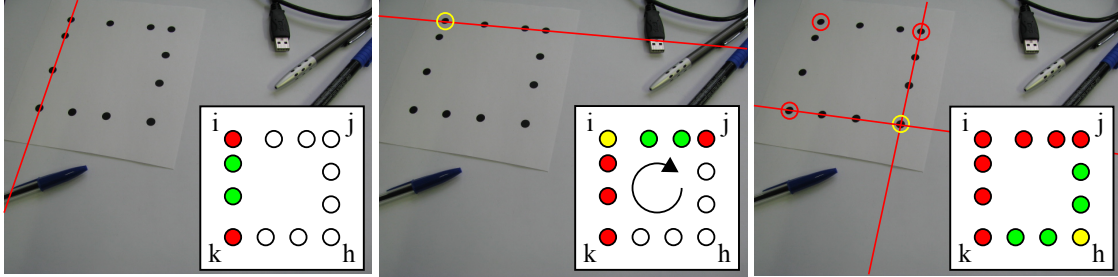
Finally, we use the cross-ratio of four collinear points A, B, C and D , a projective invariant defined as:

$$cr(A, B, C, D) = \frac{|AB|/|BD|}{|AC|/|CD|} \quad (3.1)$$

where $|AB|$ denotes the Euclidean distance between points A and B (see Figure 3.1).

The cross-ratio does not depend on the direction of the line $ABCD$, but depends on the order and the relative positions between the points. The four points can be arranged in $4! = 24$ different orderings which yield six different cross-ratios. Due to this fact, the cross-ratio is unlikely to be used directly to match a candidate set of points against a specific model, unless some information is available in order to assign a unique ordering to such points.

Many fiducial marker systems use projective and permutation P^2 -invariants [134] to eliminate the ambiguities of the different orderings. For example this invariants are used to track markers or interaction devices for augmented reality in [117] and [116]. It has to be noted, however, that permutation invariance results in the inability to estab-



(a) Search for a feasible starting side (b) Second side and corner labeling (c) Completion of the marker (if possible)

Figure 3.2: Steps of the marker detection process: in (a) a good candidate for a side is found by iterating through all the point pairs ($O(n^2)$). In (b) another connected side is searched for and, if found, the resulting angular ordering is used to label the corners found ($O(n)$). Note that the labeling is unambiguous since the corner i is associated with the lowest cross ratio. Finally, in (c) the marker is completed (if possible) by finding the missing corner among all the remaining dots. (image best viewed in colors)

lish correspondences between the detected features and points in the reference model, making it impossible to fully estimate the camera pose without relying to stereo image pairs or other features in the markers.

The main idea behind the design of the proposed Pi-Tags is to combine all the aforementioned invariants to identify each dot without ambiguities, even in presence of moderate occlusions, thus allowing fast and accurate pose estimation. To this end, it should be noted that we assume the imaging process to be projective. While this holds to a reasonable approximation with many computer vision devices with good lens and moderate focal length, wide angle cameras could hinder our assumption due to lens distortion. In this case, a proper distortion-correcting calibration step [176] should be performed before processing.

3.2.2 Marker Detection and Recognition

In our design each marker is characterized by properties that are common to all tags. Specifically, each side of the marker must be made up of exactly four dots, with the corner dots being shared and labeled as in Fig.3.2(a). For a given constant δ , a set of Pi-Tags is generated by varying the dots position constrained by the following property:

$$cr_{ij} = cr_{ik} = \delta cr_{kh} = \delta cr_{jh} \quad (3.2)$$

All these properties allow to decouple the detection and recognition pipeline into two separate steps. In the detection process a set of possible marker candidates are localized in the image by exploiting the projective invariants described in the previous section.

First, the dots are located by searching for the ellipses present in the image (projective invariance of conics). To this end we use the ellipse detector supplied by the OpenCV library [36] applied to a thresholded image. To be resilient to variations in illumination, a locally adaptive threshold is applied by [158]. Some of the ellipses found at this stage may belong to a marker in the scene (if any), others could be possibly generated by noise or clutter.

Next we group the detected ellipses into potential marker candidates. This is done considering only the centroids of the ellipses (which are a very good approximation for original circle points). The first step to gather all the points belonging to a tag is to find a viable marker side, which can be done by exploiting the straight line invariance (collinearity). For this purpose, we iterate over all the unordered pairs of dots and then, for each pair considered, we check if they are likely to be two corner points (see Fig. 3.2 a). This check is satisfied if exactly two other dots can be found lying within a fixed distance to the line connecting the first two candidate corners. The distance parameter is expressed in pixels and, since the accuracy of the estimated ellipse center is expected to be subpixel, a threshold of one or two pixels is usually enough to avoid false negatives without the risk of including misdetections. In order to obtain a better performance, this step can be accelerated using a spatial index, such as a quad-tree, rather than by testing all the ellipses found.

At this point we have identified a candidate side of the marker. Next, we validate the candidate by finding a third corner of the marker. Again, this is done by iterating over all the dots left and, for each one, by testing if it forms a candidate side with one of the current corner points (i.e. by checking that the line connecting them passes through exactly two ellipses). If a pair of sides is found then it is possible to test if they belong to a known marker and give a label to each corner. The test is carried on by verifying that the proportion between the cross-ratios of the sides is approximately 1 (in this case we are dealing with kij or jhk adjacent sides) or δ (in this case we are dealing with ijh or hki). The labeling happens by observing the ordering of the sides, which is conserved since always the same face of the tag is seen (see Fig. 3.2 b).

With two sides detected and labeled, we can recognize the marker by comparing the measured cross-ratio with the database of current markers. However, to be more robust, we search for the fourth corner with the same line-based technique. Depending on the application requirements, the search for the fourth point can be mandatory (to reduce the number of false positives and get a more accurate pose) or optional (to allow for the occlusion of at most two sides of the marker).

Once the points are detected and labeled it is possible to test if they belong to an expected marker. This final step is done by computing the average between the two or four obtained cross-ratios (divided by δ if needed) and by comparing it with all the values in the database of the tags to be searched. If the distance is below a fixed threshold, the marker is then finally recognized. Note that to avoid any ambiguity between tags, the proportion between the cross-ratios of ij sides of any pair should be different from δ .

Regarding the computation complexity of the approach, it is easy to see that find-

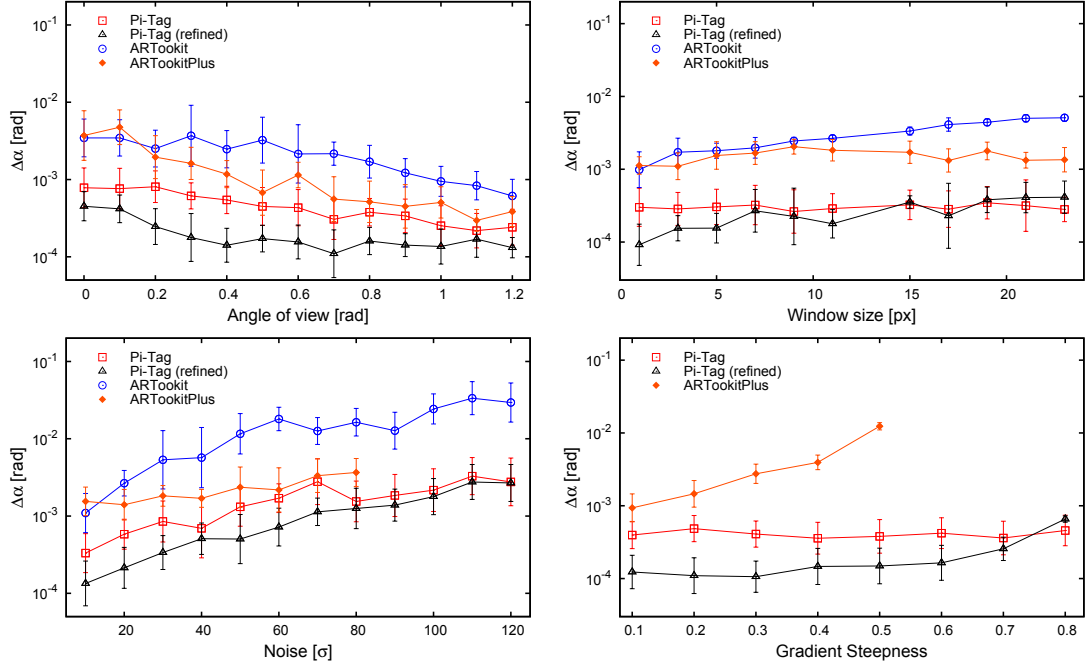


Figure 3.3: Evaluation of the accuracy of camera pose estimation with respect to different scene conditions. The first row plots the angular error as a function of view angle and Gaussian blur respectively, while the second row plots the effects of Gaussian noise (left) and illumination gradient (right, measured in gray values per image pixel). The proposed method is tested both with and without refinement. Comparisons are made with ARToolkit and ARToolkit Plus.

ing a starting side is $O(n^2)$ with the number of ellipses, while the two subsequent steps are both $O(n)$. This means that if each detected point triggers the full chain the total complexity of the algorithm could be theoretically as high as $O(n^4)$. However, in practice, given the relatively low probability of getting four ellipses in line with the correct cross ratio, most of the starting side found lead to a correct detection. In addition, even when the starting side is not correct, it is highly probable that the cross-ratio check will stop the false matching at the second step.

While a full probabilistic study would give a more formal insight, in the experimental section we will show that even with a large number of false ellipses the recognition is accurate and it is fast enough for real-time applications.

3.2.3 Estimation of the Camera Pose

Having detected and labeled the ellipses, it is now possible to estimate the camera pose. Since the geometry of the original marker is known, any algorithm that solves the PnP problem can be used. In our tests we used the *solvePnP* function available in OpenCV. However, it should be noted that, while the estimated ellipse centers can be

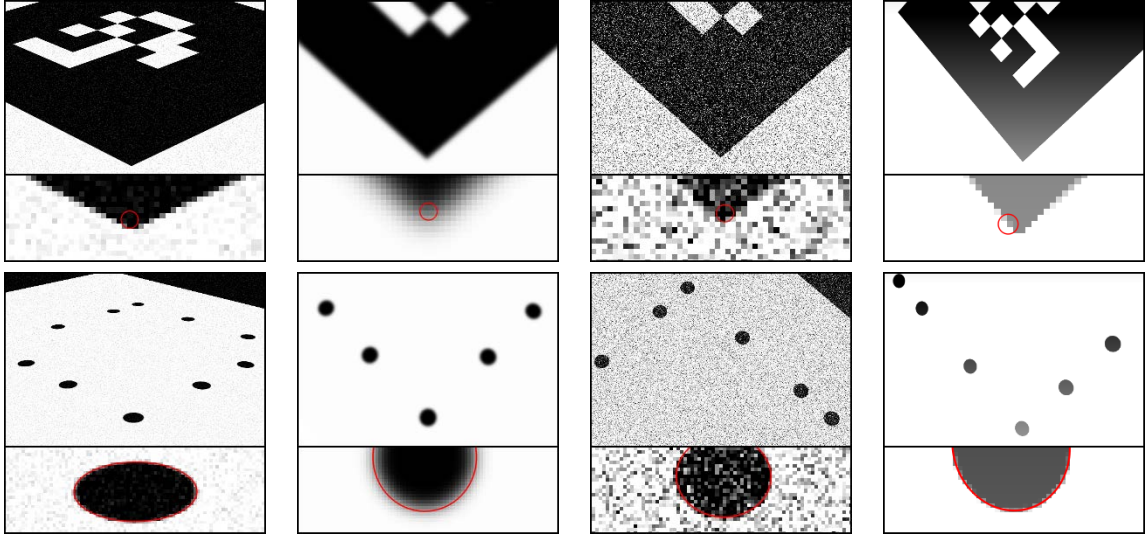


Figure 3.4: Some examples of artificial noise used for synthetic evaluation. The artificial noise is, respectively, light Gaussian noise at grazing view angle (first column), blur (second column), strong Gaussian noise (third column) and illumination gradient (fourth column). The tested markers shown are ARToolkit Plus (first row) and Pi-Tag (second row).

good enough for the detection step, it is reasonable to refine them in order to recover a more accurate pose. Since this is done only when a marker is found and recognized, the computational cost is limited. In our experiments we opted for the robust ellipse refinement approach presented in [147].

In addition, to obtain a more accurate localization, one might be tempted to correct the projective displacement of the ellipses centers. However, according to our tests, such correction in general gives little advantage and sometimes leads to a slightly reduction in accuracy. Finally, we also tried the direct method outlined in [101], but we obtained very unstable results, especially with small and skewed ellipses.

3.3 Experimental Validation

In this section we evaluate the accuracy and speed of the Pi-Tag fiducial markers and compare them with ARToolkit and ARToolkitPlus.

A first batch of tests is performed with synthetically generated images under different condition of viewing direction, noise, and blur. This allows us to compare the different techniques with a perfect ground truth for the camera pose, so that even slight differences in precision can be detected. The accuracy of the recovered pose is measured as the angular difference between the ground truth camera orientation and the obtained pose. While this is a subset of the whole information related to the pose, this is an important parameter in many applications and allows for a concise analysis.

A second set of experiments is aimed at characterizing the behaviour of Pi-Tags with respect to its resilience to occlusion, the presence of false positives, and the sensitivity to the threshold parameters, as well analyze computational time required by the approach.

Finally, four real-world application of the proposed tag are studied. Namely, we show the effectiveness of these markers as tools for contactless measurement, camera calibration, and 3D surface alignment. In addition, we also describe a possible use of Pi-Tags with non-square aspect ratio for projected augmented reality applications.

The implementations of ARToolkit and ARToolkitPlus used are the ones freely available at the respective websites. The real images are taken with a 640x480 CMOS webcam for the occlusion test and with a higher resolution 1280x1024 CCD computer vision camera with a fixed focal length lens for the measurement tests.

All the experiments have been performed on a typical desktop PC equipped with a 1.6Ghz Intel Core Duo processor and 2GB of RAM.

3.3.1 Accuracy and Baseline Comparisons

In Fig. 3.3 the accuracy of our markers is evaluated. In the first set of experiments the marker is tested at increasing grazing angles and with a minimal additive Gaussian noise. It is interesting to note that oblique angles lead to a higher accuracy for all the methods, as long as the markers are still recognizable. This is explained observing that large angles of view constraint of the reprojections of the points to the image plane more than almost orthogonal views. Pi-Tag shows better results both when the pose is evaluated with the original thresholded ellipses and after the refinement.

In the second test we evaluated the effects of Gaussian blur, which appears to have a limited effect on all the techniques. This is mainly related to the fact that all methods perform a preliminary edge detection step, which in turn applies a convolution kernel. Hence, it is somewhat expected that an additional blur does not affect much the marker localization. In the third test an additive Gaussian noise was added to images with an average view angle of 0.3 radians and no artificial blur was added.

The performance of all methods decreases with increasing levels of noise and ARToolkitPlus, while in general more accurate than ARToolkit, breaks when dealing with a noise with a standard deviation greater than 80 (pixel intensities goes from 0 to 255). Finally, the effect of illumination gradient is tested only against ARToolkitPlus (since ARToolkit cannot handle this kind of noise), which, again, exhibits lower accuracy and breaks with just moderate gradients.

Overall, these experiments confirm that Pi-Tag outperforms the alternative marker systems. This is probably due both to the higher number of pinpointed features and to the better accuracy attainable using circular patterns rather than corners [129].

In practical terms, the improvement is not negligible. In fact an error as low as 10^{-3} radians still produces a jitter of 1 millimetre when projected over a distance of 1 meter. While this is a reasonable performance for augmented reality applications, it is unacceptable for precise contactless measurements.

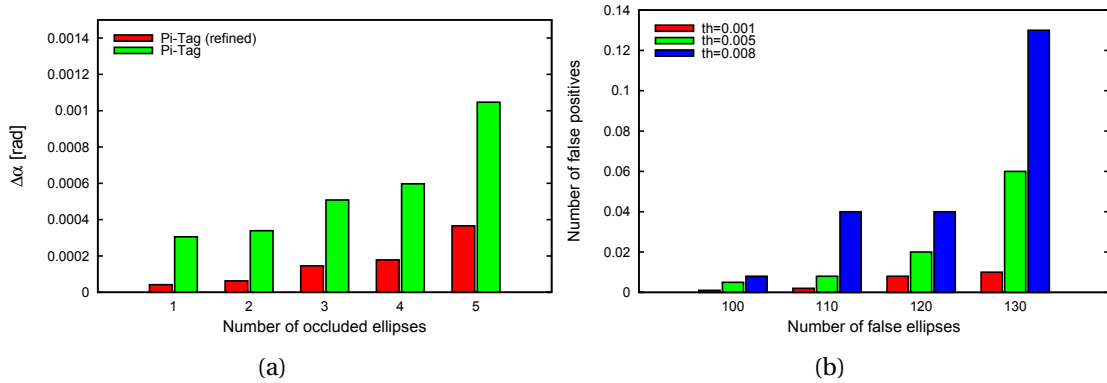


Figure 3.5: Left (a): Evaluation of the accuracy of the estimated camera pose when some dot of the marker are occluded (note that if more than 5 dots are missing the marker is not detected). Right (b): Evaluation of the number of false positive markers detected as a function of the number of false ellipses introduced in the scene and the threshold applied to the cross-ratio.

3.3.2 Resilience to Occlusion and False Ellipses

One of the characteristics of Pi-Tag is that it can deal with moderate occlusion. In Fig. 3.5(a) we show how occlusion affects the accuracy of the pose estimation (i.e., how well the pose is estimated with a subset of the dots, regardless to the possibility of recognizing the marker with those dots).

While we observe a decrease in the accuracy as we increase the occlusion, the precision is still acceptable even when almost half of the of the dots are not visible, especially for the refined version of the tag. In Fig. 3.5(b) we evaluate the proportion of false marker detections obtained by introducing a large amount of false ellipses at random position and scale. When the threshold on the cross-ratio is kept tight it is possible to obtain a very low rate of false positives even with a large number of random dots.

3.3.3 Performance Evaluation

Our tag system is designed for improved accuracy and robustness to occlusion rather than for high detection speed. This is quite apparent in Fig. 3.6(a), where we can see that the recognition could require from a minimum of about 10 ms (without false ellipses) to a maximum of about 150 ms.

By comparison, ARToolkit Plus is about an order of magnitude faster [185]. However, it should be noted that, despite being slower, the frame rates reachable by Pi-Tag (from 100 to about 8/10 fps) is still sufficient for real-time applications (in particular when few markers are viewed at the same time). Further, our code is not as heavily optimized as ARToolkitPlus, which gained a factor of 10 performance gain with respect to ARToolkit. It is reasonable to assume that a similar optimization effort would result

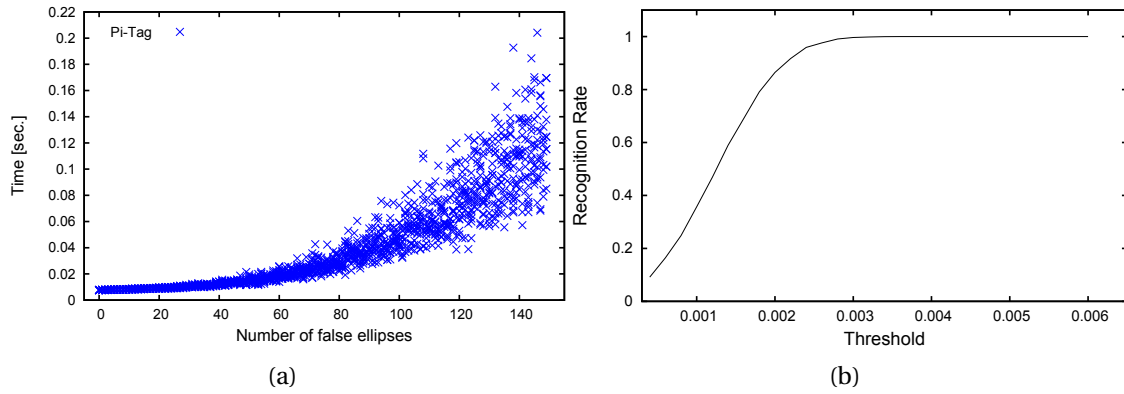


Figure 3.6: Left (a): Evaluation of the detection and recognition time for the proposed marker as random ellipses are artificially added to the scene. Right (b): Evaluation of the recognition rate achieved on a real video of about ten minutes in length, with respect to different thresholds applied to the cross-ratio.

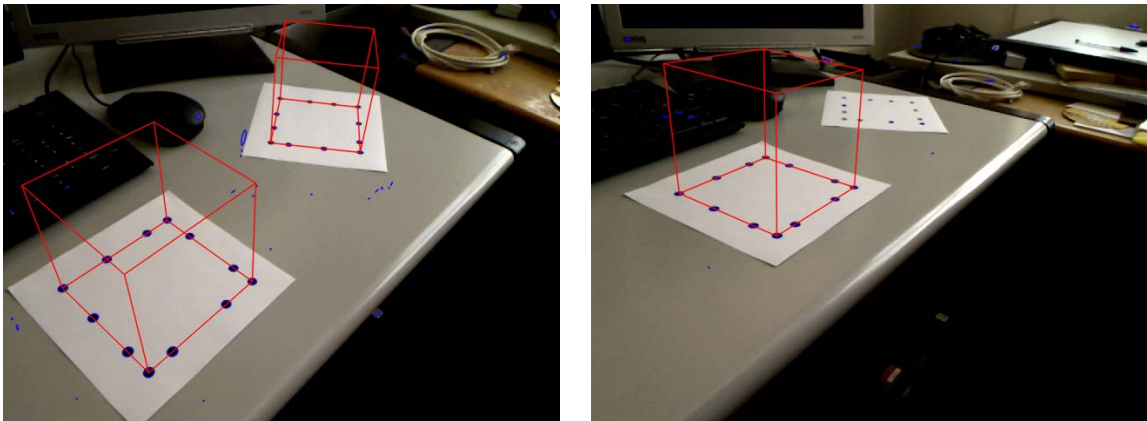


Figure 3.7: Recognition fails when the marker is angled and far from the camera as the ellipses detectors cannot detect the circular features.

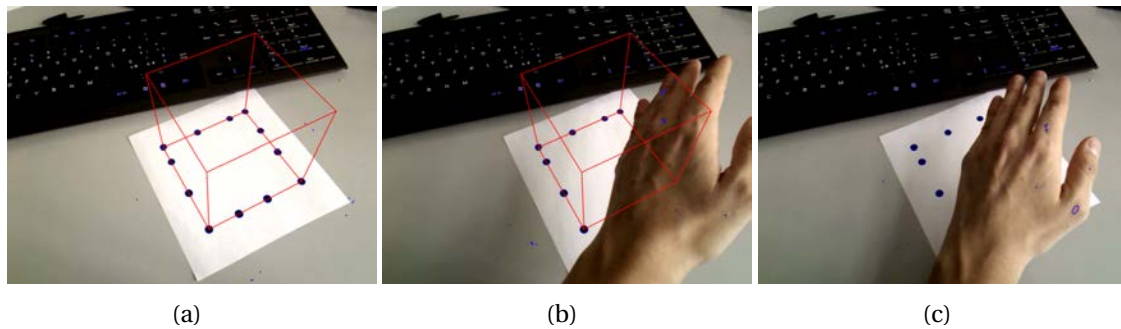


Figure 3.8: Some examples of the behaviour in real videos: In (a) the marker is not occluded and all the dots contribute to the pose estimation. In (b) the marker is recognized even if a partial occlusion happens. In (c) the marker cannot be detected as the occlusion is too severe and not enough ellipses are visible.

in a similar gain in performance.

3.3.4 Behavior on Real Videos

In addition to the evaluation with synthetic images, we also performed some qualitative and quantitative tests on real videos. In Fig. 3.8 some experiments with common occlusion scenarios are presented. Note that when at least two sides are fully visible the marker is still recognized and the correct pose is recovered.

In Fig. 3.6(b) we plot the recognition rate of the markers as a function of the cross-ratio threshold. this was computed from a ten minute video presenting several different viewing conditions. It is interesting to note that even with a small threshold we can obtain a complete recall (compare this with the threshold in Fig. 3.5(b)).

Finally, Fig. 3.7 highlights an inherent shortcoming of our design: The relatively small size of the base features may result in a failure of the ellipse detector when the tag is far away from the camera or very angled, causing the dots to become too small or to blended together.

3.3.5 Using Pi-Tag for camera calibration

Camera calibration is a fundamental task whenever imaging devices are to be used in measurement applications. In fact, if the intrinsic parameters of the devices (and thus the image formation process) are not known with high accuracy, it is not possible to relate the points on the image plane to the phenomena that generated them. In the case of Pi-Tags, detection and recognition entirely happen in the image plane, for this reason calibration is not needed per se. However, a calibration procedure based solely on Pi-Tags provides a useful testbed.

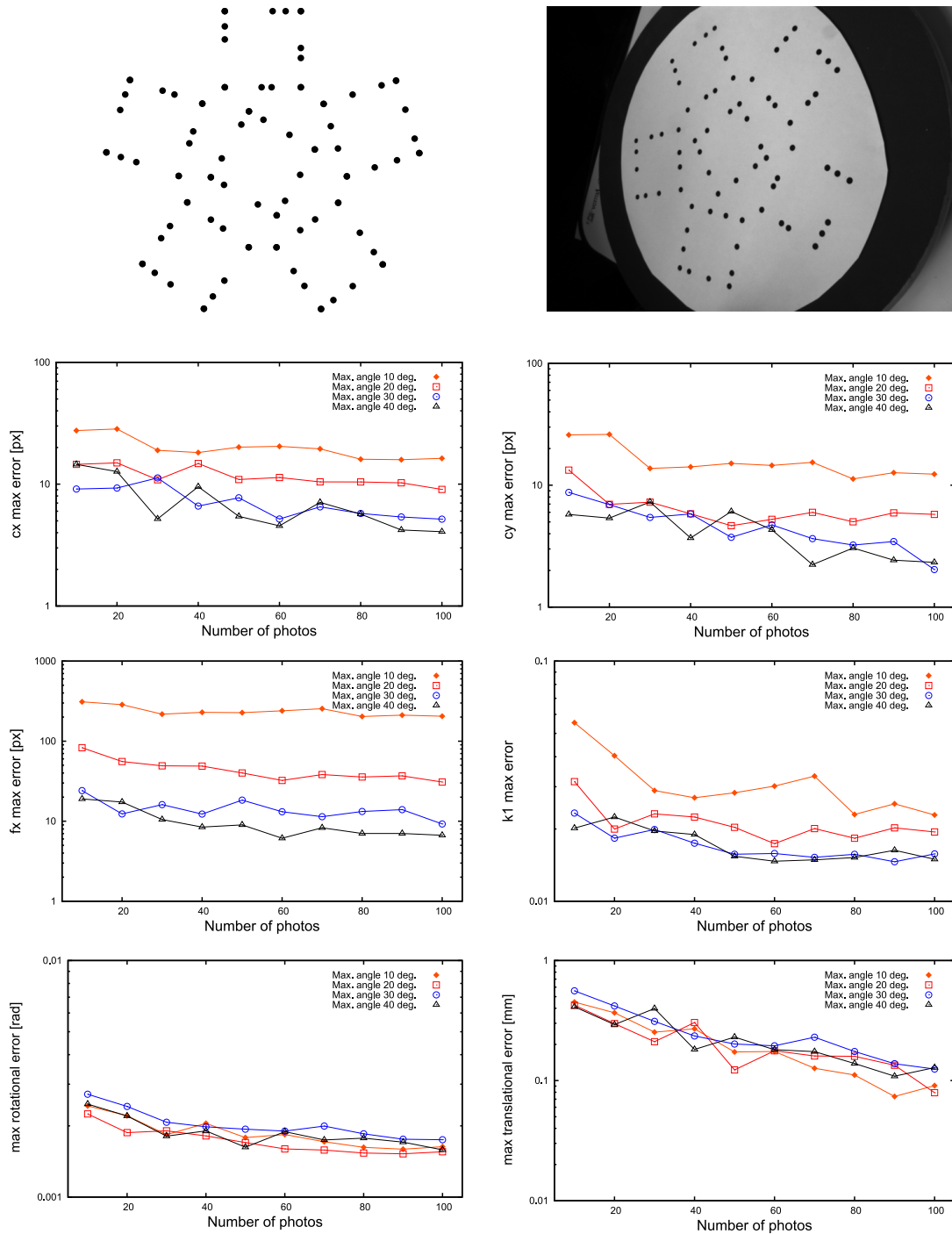


Figure 3.9: Evaluation of the quality of mono and stereo calibration obtained using Pi-Tags as fiducial markers.

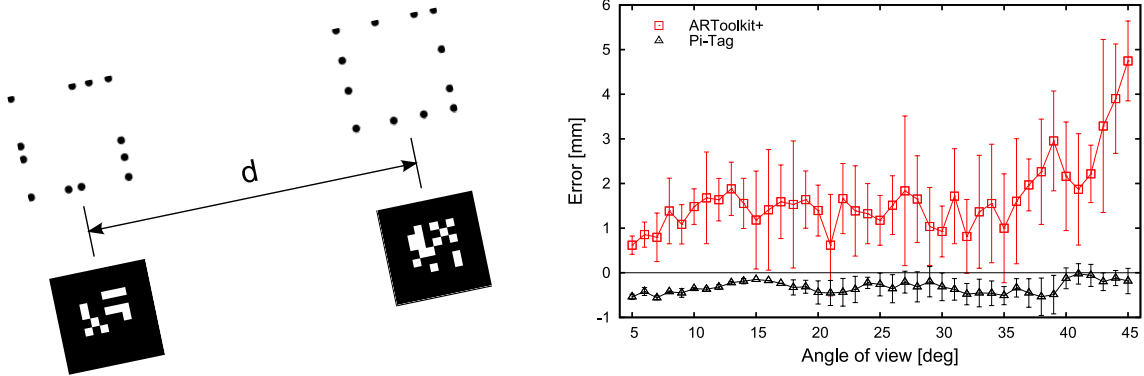


Figure 3.10: Performance of the proposed fiducial marker as a tool for image-based measurement.

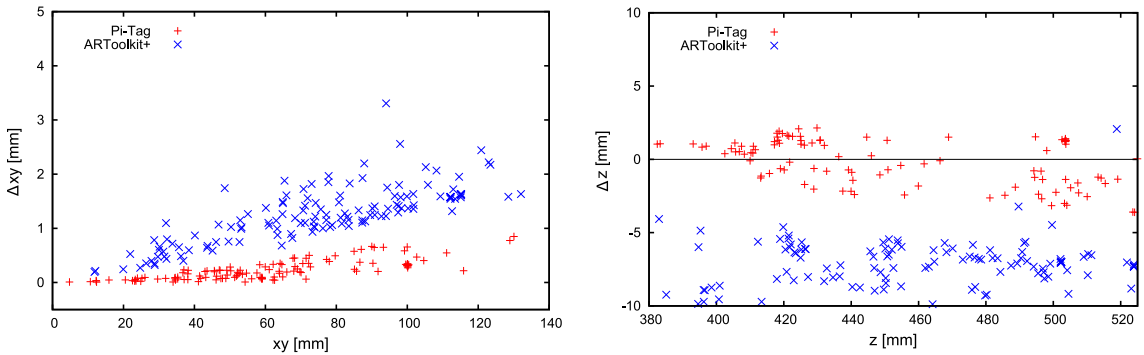


Figure 3.11: Analysis of the measurement error committed with respect to different positions of the marker pair.

There are many different image formation models and of course each one comes with a different set of parameters. In the following tests we adopted the model proposed by [88]. In this model the imaging device is represented as a pinhole camera whose incoming rays are displaced on the image plane through a polynomial distortion. Such distortion is parametrized by three coefficient of the polynomial usually labeled k_1 , k_2 and k_3 , being k_1 the most relevant (in terms of displacement) and k_3 the least relevant.

Once the distortion is factored out, the pinhole part of the model is defined through the principal point (i.e. the projection on the image plane of the projective center) labeled as (cx, cy) and the focal length (fx, fy) (which are two parameters to account for non-square pixels). Since we are dealing with low distortion cameras that are equipped with sensors with square pixels, we considered only the first distortion coefficient k_1 and we assumed $fx = fy$.

The first set of calibration experiments was performed using a target made up of Pi-Tags placed according to a known geometry and printed on an A3 sheet with a standard inkjet printer (see Fig. 3.9). Several shots of the target were captured with differ-

ent viewing angles and at different distances. For each shot the tags were detected and recognized, and an association between the 2D points on the image plane and the 3D point of the known model was stored. This data was finally fed to the camera calibration procedure available in the OpenCV library. Since both the target viewing angle and the number of shots are relevant factors for the quality of the calibration we studied the effect of both.

In the first four graphs of Fig. 3.9 we show the absolute distance between the parameters recovered with the described procedure and a ground truth calibration performed with a full checkerboard pattern with about 600 reference corner and using 200 shots. Specifically, for each number of shots and maximum angle we selected 30 random set of images to be used for calibration.

The distance plotted in the graph is the maximum absolute error committed in the 30 calibrations. From these graphs it is possible to see that taking shots with a large enough viewing angles is important. This is due both to the stronger constraint offered to pinhole parameters by angled targets, and to the more accurate pose estimation offered by Pi-Tag when the angle of view is not negligible (see Fig. 3.3). In addition we can also observe that taking a large number of samples increases monotonically the accuracy obtained.

In the second set of calibration experiments the tags were used to estimate the relative pose between two cameras of known intrinsic model. This stereo calibration is useful in many reconstruction tasks where the epipolar geometry between more than one camera can be exploited to fully localize the 3D points that are imaged (see [86]).

Again, we estimated a ground truth relative pose between a pair of identical fixed cameras using a specialized target and plotted on the bottom row of Fig. 3.9 the maximum absolute error between the ground truth and the values obtained in 20 calibrations performed on randomly selected shots with a given maximum viewing angle. In this condition the viewing angle is less important, but still a large number of shots gives better results.

3.3.6 Contactless measurements

A calibrated camera can be used in conjunction with any detectable marker of a known size as a contactless measurement tool. To assess the precision offered for this use scenario, we printed two Pi-Tags and two ARToolkitPlus tags at a distance (center to center) of 200 millimetres (see Fig. 3.10). Subsequently, we took several shots and estimated such distance. In the graph displayed in Fig. 3.10 we plotted the difference (with sign) between the measured and real distance between tags at several viewing angles. Pi-Tag consistently exhibits smaller errors and a smaller variance. As usual, the measure improves slightly as the viewing angle increases. It is interesting to note that, according to our measurements, Pi-Tag has a tendency to underestimate the distance slightly, while ARToolkitPlus do exactly the opposite.

In Fig. 3.11 we show two scatter plots that depict respectively the error in localization on the x/y plane (as the norm of the displacement vector) with respect to the po-

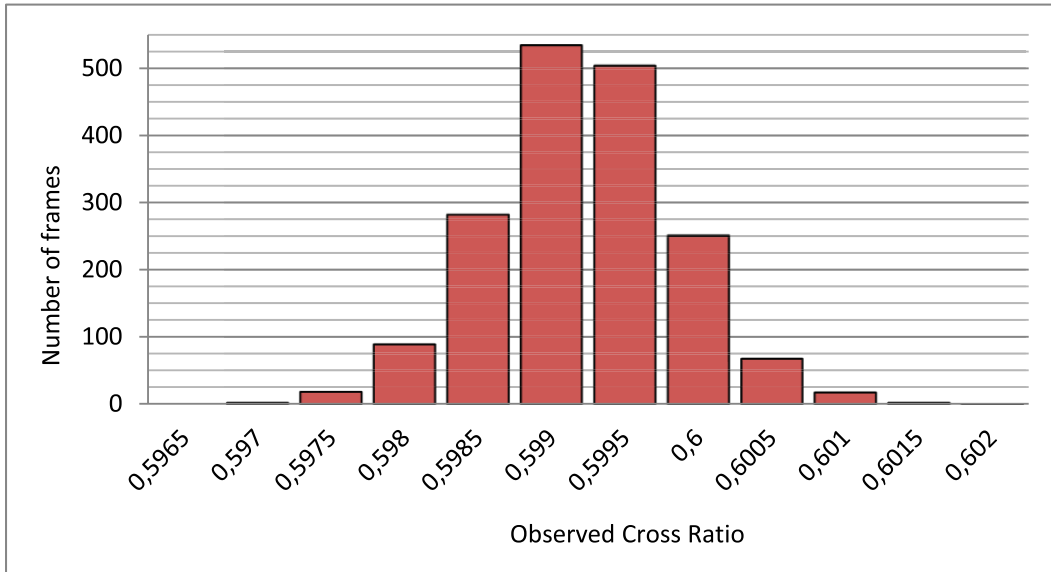


Figure 3.12: Cross ratios measured in a real video sequence.

sition of the target and the difference in depth estimation (signed) with respect to the depth of the target. In this case, the ground truth was obtained using a stereo camera pair properly calibrated. Also in this test Pi-Tag obtains better results than ARToolkit-Plus. The larger error in the localization near the image border is probably due to the inability of the polynomial distortion model to fully capture the pixel displacement far away from the principal point. Also, the spread of the error in estimating the depth is larger when the target is far from the camera, which is expected as the resolution of the detected ellipses decreases and so does the accuracy in the location of their centers.

3.3.7 Scalability over the number of markers

In many practical scenarios it could be useful to place a large number of markers in the scene. For this reason, it is important to assess the ability of the proposed approach to generate markers distinctive enough to avoid wrong classifications even with big databases. To this end, we first evaluated the distribution of the measured cross-ratio in a real video sequence of about 2000 frames showing a marker under various angles and lighting conditions. As shown in Fig. 3.12, the acquired cross-ratio appears to behave as a Gaussian distributed random variable.

If we deem this model as reasonable, it is easy enough to estimate both the probability of missing a marker and of a wrong classification (see Fig. 3.13). Given the standard deviation of the measured cross-ratio σ_{cr} (that we assume uniform over all the database) and a tolerance ϵ between the acquired value and the target cross-ratio cr , the probability of a false negative for a given marker is exactly:

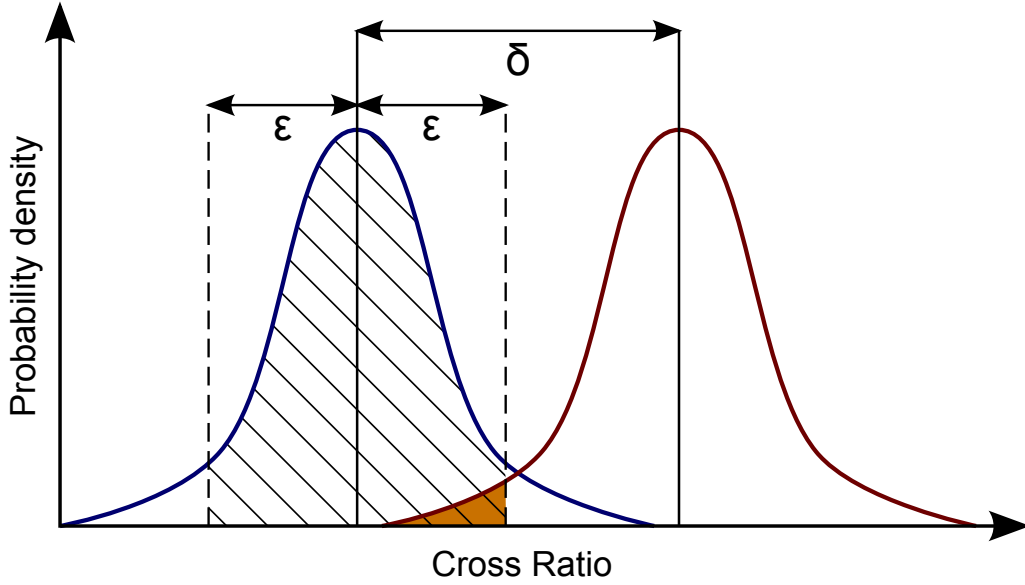


Figure 3.13: Relation between recognition margin and cross-ratio separation among markers.

$$1 - \int_{cr-\epsilon}^{cr+\epsilon} \frac{1}{\sigma_{cr}\sqrt{2\pi}} e^{-\frac{(x-cr)^2}{2\sigma_{cr}^2}} dx \quad (3.3)$$

In addition, given a minimum separation between cross-ratios in the database of δ (assumed to be bigger than ϵ), the probability of a wrong classification is upper bounded by:

$$2 \int_{-\infty}^{cr-(\delta-\epsilon)} \frac{1}{\sigma_{cr}\sqrt{2\pi}} e^{-\frac{(x-cr)^2}{2\sigma_{cr}^2}} dx \quad (3.4)$$

By choosing apt values for ϵ and δ it is possible to set the sought balance between an high recognition ability and a low number of misclassifications. For instance, as the measured standard deviation in our test video was $\sigma_{cr} = 6 \cdot 10^{-4}$ a choice of $\epsilon = 2 \cdot 10^{-3}$ would grant an expected percentage of false negative lower than 0.1%. At the same time, a choice of $\delta = 4 \cdot 10^{-3}$ would set the rate of wrong classifications below 0.01%.

To translate these numbers into a feasible database size, it is necessary to account for the physical size of the marker and of the dots. In our test video we used a square marker with a side of 10cm and with a dot diameter of 1cm.

Within these conditions we were able to obtain cross-ratios from 0.026 to 1.338 keeping enough white space between dots to make them easily detectable by the camera. Assuming that the cross-ratios in the database are produced to be evenly distributed, a span of about 1.3 grants for a total of about 300 distinct markers with the above mentioned levels of false negatives and wrong classifications.

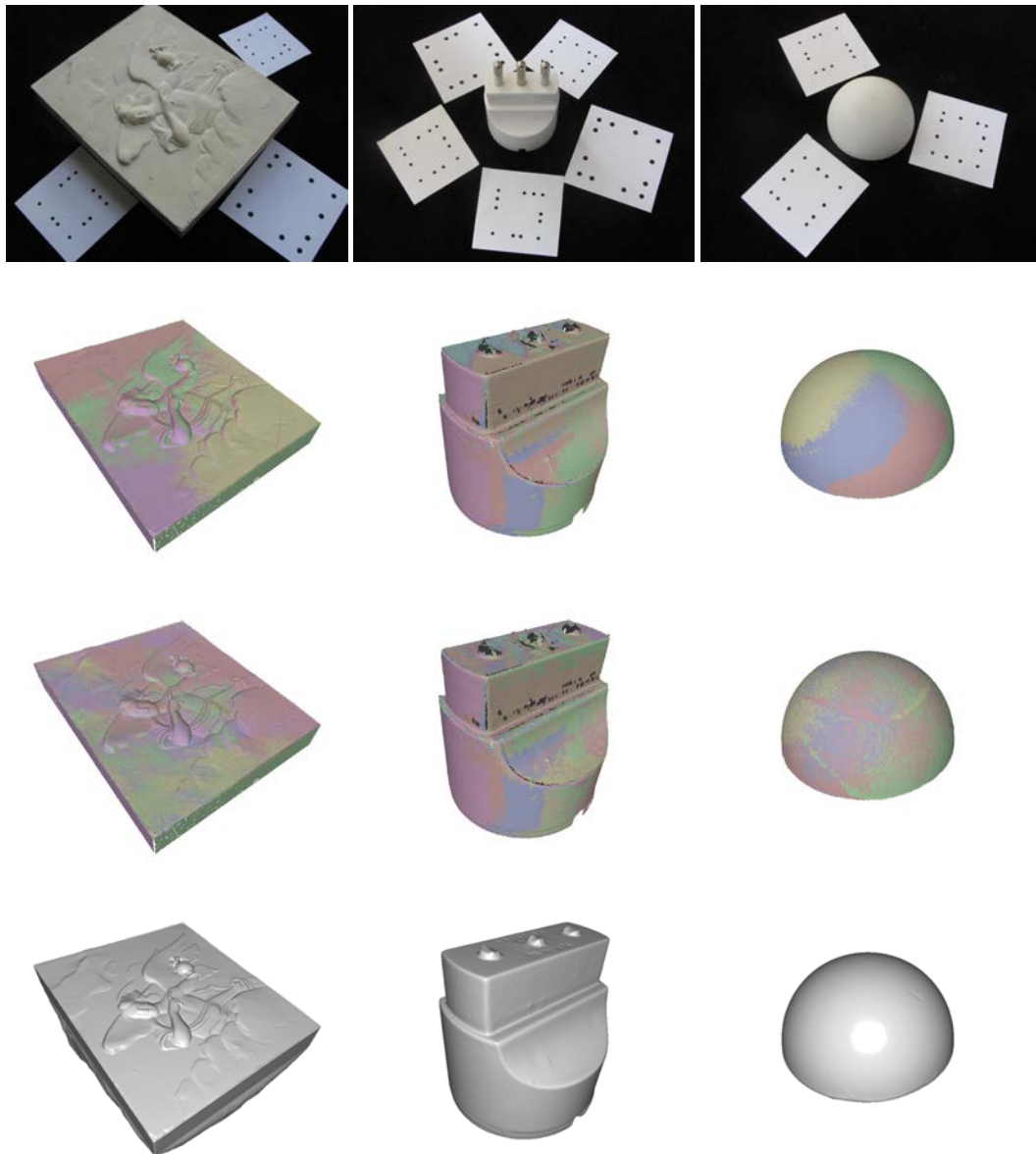


Figure 3.14: Examples of surface reconstructions obtained by acquiring several ranges with a structured-light scanner and by using Pi-Tag markers to set a common reference (image best viewed in color).



Figure 3.15: Actual setup and examples of usage by moving the controller above the printed map.

3.3.8 Registration of 3D surfaces

In order to further evaluate the utility of the new marker design in practical scenarios we performed a last set of qualitative experiments. To this end, we captured several range images from different objects using a 3D scanner based on structured light. These objects were placed on a turntable and surrounded with Pi-Tags (see Fig. 3.14). During each acquisition step we took an additional shot that captures the Pi-Tags in natural lighting, thus allowing us to use them to recover the pose of the object on the turntable. This is a typical application of artificial markers, since most algorithms used to align 3D surfaces need a good initial motion estimation to guarantee a correct convergence. In the second row of Fig. 3.14 we show the overlap of several ranges using the pose estimated with Pi-Tags, without any further refinement. In the third row the initial alignment is refined using a state-of-the-art variant of the well-known ICP algorithm (see [156]). After the refinement, slight improvements in the registration can be appreciated, especially in the “hemisphere” object. The smooth blending of colors obtained means that ICP was able to obtain a good alignment, which in turn testifies the quality of the initial pose estimated using the Pi-Tags. In the last row we present the watertight surface obtained after applying the Poisson surface reconstruction algorithm [105] to the aligned range images. Overall, the surfaces are smooth and do not exhibit the typical artefacts related to misalignment. In addition, the fine details of the first object are preserved.

3.3.9 Applications in Projected Augmented Reality

An interesting property of the proposed tag design is that there is no need for it to be square. In fact, any aspect ratio can be adopted without compromising the properties of the cross ratio that are needed for the recognition of the tag and for the pose estimation. We combined this exact property with the fact that the inside of the tag is blank (as with other frame-based designs [184]) to build an additional application within the domain of the Projected Augmented Reality.

Specifically, we built a system where a Pi-Tag is used both as a passive input device

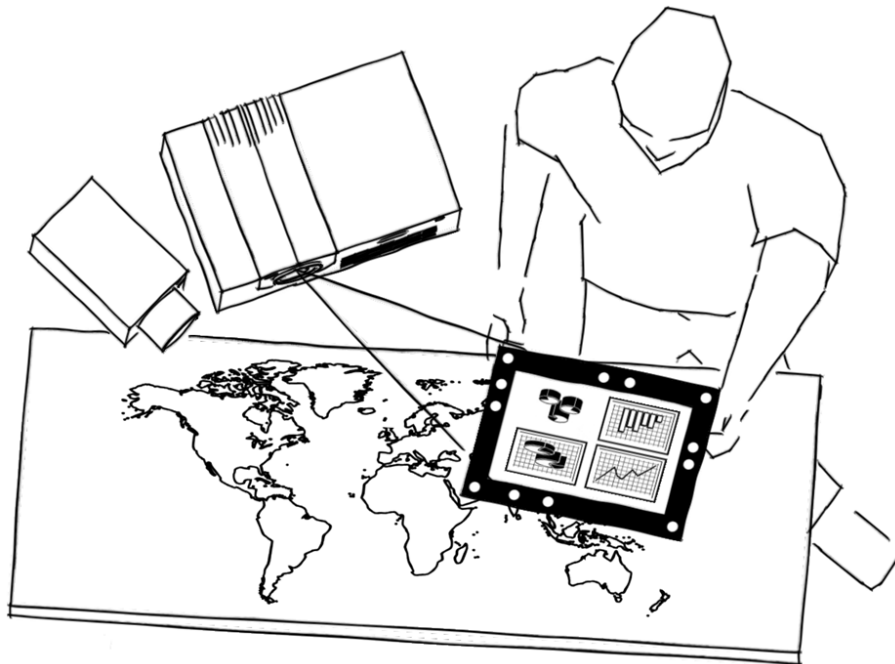


Figure 3.16: A schematic representation of the setup.

and as a display. The input device can be regarded as a 3D mouse that can be used to explore interactive content. The ability to display data on the tag is obtained by using an external projector that is aimed toward the white surface internal to the tag.

A schematic representation of the setup can be seen in Fig. 3.16. From a technical stand-point the system is made up of a digital projector, a camera, a board where the map of interest is physically printed and a navigation device. The projector and the camera are rigidly mounted on a stand and are both oriented toward the table so that their frustum covers the entire area of interest. The navigation device is basically a rectangular rigid board that exhibits a white matte projection area and a frame that contains the fiducial marker to track. Since the rigid transform that binds the camera to the projector is known and the projector frustum itself corresponds to the map area, all the parameters are available to reconstruct the position of the navigation device with respect to the map and to the projector and thus to display on the matte area some contextual data related to the location observed by the user.

The geometrical relation between the projector and the navigation device is used to rectify the displayed image so that it appears exactly as if it was formed on the screen of an active device. By printing different markers, more than one navigation device can be used simultaneously, thus allowing many users to operate on the table. Finally, since the marker position is determined in 3D, additional functions such as zooming can be controlled through the vertical position of the device. In Fig. 3.15 an actual implementation of the setup and the zooming effect attainable are shown.

The main challenge of the projection calibration is to estimate its projection matrix

$P = K_p[R_p|T_p]$, where

$$\mathbf{K}_p = \begin{bmatrix} f x_p & 0 & c x_p \\ 0 & f y_p & c y_p \\ 0 & 0 & 1 \end{bmatrix}$$

are projector intrinsic parameters, and $[R_p|T_p]$ is the relative pose of the projector with respect to the marker, or the extrinsic parameters. Once the matrix P has been estimated, a 3D point p_m lying on the marker plane can be projected by transforming its 3D coordinates $[x_w y_w 0]^T$ to projector image-space pixel coordinates $[u_p v_p]^T$ with the following equation:

$$\mathbf{K}_p = \begin{bmatrix} u_p \\ v_p \\ 1 \end{bmatrix} = P \begin{bmatrix} x_w \\ y_w \\ 0 \\ 1 \end{bmatrix} = P p_w$$

Unfortunately, the projector cannot estimate the relative pose $[R_p|T_p]$ by itself because it is a pure output device. To provide that data, a camera is placed nearby ensuring that the viewing frustum of the projector is contained in the viewing frustum of the camera. As long as the relative position between the camera and projector remains unchanged, $[R_p|T_p]$ can be estimated in terms of the camera pose $[R_c|T_c]$ obtained via fiducial markers in the following way:

$$\begin{bmatrix} R_p & T_p \\ \vec{0} & 1 \end{bmatrix} = \begin{bmatrix} R_c p & T_c p \\ \vec{0} & 1 \end{bmatrix} \begin{bmatrix} R_c & T_c \\ \vec{0} & 1 \end{bmatrix}$$

The estimation of K_p and $[R_{cp}|T_{cp}]$ can be obtained from a set of known 3D-2D correspondences as in subsection 3.3.5, however, as the projector cannot "see" the markers and retrieve 3D positions of dots in the calibration target, an alternative method is used to provide this mapping. A big square Pi-Tag marker is printed on a planar surface and placed under the camera/projector frustum (Fig. 3.17). Once the tag is placed, a snapshot is taken by the camera and used for background subtraction. This allow us to project a dot with the projector by randomizing its 2D position in projector plane, and detect its center with no ambiguity using the camera. If the camera detects that the projected dot lies inside the marker, the 3D position of the dot can be recovered because the marker plane position is known with respect to the camera via Pi-Tag pose estimator. The whole process can be summarized as follows:

1. A planar surface with a Pi-Tag marker is placed randomly under camera/projector frustum, and a snapshot is taken.
2. A dot $p_p = [u_p v_p]^T$ is projected randomly by the projector. Via background subtraction the camera can identify the dot projected and determine its 2D position $p_c = [u_c v_c]^T$ in the camera image plane.

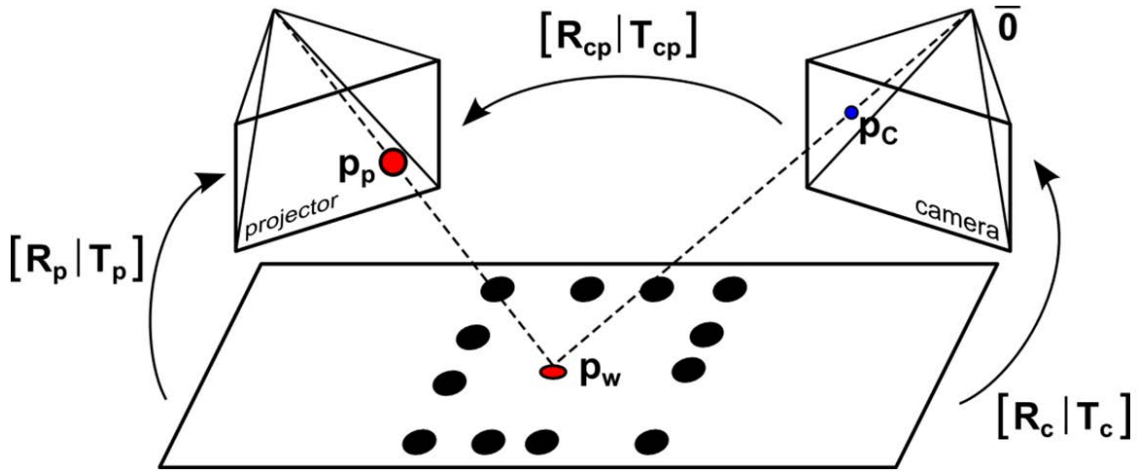


Figure 3.17: Geometric relation between the entities involved in the projector calibration procedure.

3. If the 2D position of the dot lies inside the marker, its 3D position $p_w = [x_w y_w z_w]^T$ (in camera world) can be recovered as the intersection of the line from the camera center of projection $\vec{0}$ and the point $[\frac{u_c - cx_c}{f_{x_c}} \frac{v_c - cy_c}{f_{y_c}} 1]^T$ and the marker plane, computed using Pi-Tag pose estimator.
4. Steps 2 and 3 are repeated to collect hundreds of 3D-2D correspondences (p_w, p_p) from this point of view
5. Steps 1 to 4 are repeated to collect correspondences between different point of views. For our purposes, about half a dozen of different point of views is usually enough.
6. OpenCV `calibrateCamera` function is used to estimate K_p and the rigid motion $[R_{cpi} | T_{cpi}]$ between the randomly-projected 3D points in camera world from each point of view and the projector. As final $[R_{cp} | T_{cp}]$ we simply choose the rigid motion with respect to the first point of view $[R_{cp0} | T_{cp0}]$ but different strategies may be used.

Only the first step requires human intervention instead of points 2 and 3 that needs to be iterated thoroughly to collect a large set of correspondences. Even if the process is automatic, steps 2 and 3 may require a very long time depending by the probability that the random dot p_p will lie inside the marker at each iteration. To speed up the calibration procedure, for each point of view, after at least 4 projections lying inside the marker, an homography H can be computed that maps points from camera image plane to projector image plane. With the homography H , each point p_p can be randomized directly lying inside the marker thus eliminating the waste of time required to guess the correct set of positions. In our setup we are able to collect more than 10 correspondences per second, for an average calibration time of less than 15 minutes.

3.4 Conclusions

The novel fiducial marker proposed in this chapter exploits the interplay between different projective invariants to offer a simple, fast and accurate pose detection without requiring image rectification. Our experimental validations shows that the precision of the recovered pose outperforms the current state-of-the-art. In fact, even if relying only on a maximum on 12 dots, the accuracy achieved by using elliptical features has been proven to give very satisfactory results even in presence of heavy artificial noise, blur and extreme illumination conditions. This accuracy can be further increased by using an ellipse refinement process that takes into account image gradients. The marker design is resilient to moderate occlusion without severely affecting pose estimation accuracy. The internal redundancy exhibited by its design allows to compensate the strongly non-uniform distribution of cross-ratio and also permits a good trade-off between the recognition rate and false-positives. Even taking into account the limited number of discriminable cross-ratios, the design still offers a reasonable number of distinct tags. Further, the proposed design leaves plenty of space in the marker interior for any additional payload. Since it works entirely in image-space, our method is affected by image resolution only during the ellipse detection step, and is fast enough for most real-time augmented reality applications.

Of course those enhancements do not come without some drawbacks. Specifically, the small size of the circular points used can lead the ellipse detector to miss them at great distance, low resolution, or if the viewing point is very angled with respect to the marker's plane. This limitations can be partially overcome by increasing the ratio between the size of the ellipses and the size of the marker itself, thus limiting the range of possible cross-ratio values and the total number of different tags that can be successfully recognized.

4

Robust Fiducial Marker Based on Cyclic Codes

In this chapter we introduce a general purpose fiducial marker which exhibits many useful properties while being easy to implement and fast to detect. The key ideas underlying our approach are three. The first one is to exploit the projective invariance of conics to jointly find the marker and set a reading frame for it. Moreover, the tag identity is assessed by a redundant cyclic code defined through the same circular features used for detection. Finally, the specific design and feature organization of the marker are well suited for several practical tasks, ranging from camera calibration to holding information payloads.

4.1 Introduction

In this chapter we introduce a novel fiducial marker system that takes advantage of the same basic features for detection and recognition purposes. The marker is characterized by a circular arrangement of dots at fixed angular positions in one or more concentric rings. Within this design, the projective properties of both the dots and the rings they compose are exploited to make the processing fast and reliable. In the following section we describe the general nature of our marker, the algorithm proposed for its detection and the coding scheme to be used for robust recognition. In the experimental section we validate the proposed approach by comparing its performance with two widely used marker systems and by testing its robustness under a wide range of noise sources.

4.2 Rings of UNconnected Ellipses

We design our tags as a set of circular high-contrast features (*dots*) spatially arranged into concentric *levels* (See Fig. 4.1). The tag internal area, delimited by the outer level, is divided into several evenly distributed circular *sectors*. Each combination of a level and a sector defines a *slot* that may or may not contain a dot.

In a tag composed by m levels and n sectors, we can encode a sequence of n symbols taken from an alphabet of 2^m elements. Each element of the alphabet is simply defined as the number binary encoded by the presence or absence of a dot. For example, if the 14th sector of a 3-levels tag contains a dot in the first and the last level, we encode the 14th symbol with the number $5_{10} = 101_2$. In this chapter we propose two instances of such design, namely *RUNE-43* and *RUNE-129*. The first is composed by a single level divided into 43 slots. Since the alphabet contains only 2 elements, each *RUNE-43* encodes a sequence of 43 binary symbols. Conversely, the latter is composed by 3 levels divided into 43 sectors. Three slots for each sector allow to encode a sequence of 43 symbols from an alphabet of $2^3 = 7$ elements. Not surprisingly, not all the possible codes can be used as valid codes for the tag. For instance, the tag composed by only empty slots does not make any sense. Therefore, we require the coding strategy to respect some properties to uniquely identify each dot regardless the projective transformation involved. We discuss this topic in detail in section 4.2.3.

Finally, we set the dot radius equals to κ -times to the radius of the level at which the dot is placed. We can take advantage of this property to dramatically speed up the detection as explained in section 4.2.1.

4.2.1 Candidate selection with a calibrated camera

One of the core features of every fiducial marker system is its ease of detection. Even if one of our principles is to promote accuracy over speed, we still need to setup an efficient way to identify each circular feature among the tags. Given an image, we start by

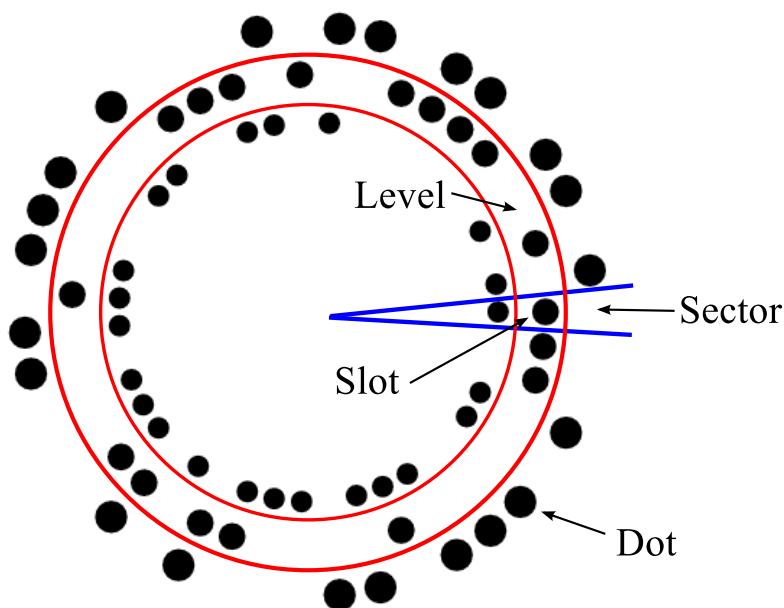


Figure 4.1: Our proposed marker design divided into its functional parts. An instance of a RUNE-129 with 3 levels is displayed.

extracting a set of candidate dots. To do this, we use a combination of image thresholding, contour extraction and ellipse fitting provided by the OpenCV library. Additionally, a subsequent naive filtering step based on dot eccentricity and area keeps only features respecting a reasonable prior. Finally, each extracted ellipse can be further refined by using common sub-pixel fitting techniques such the one proposed in [147]. We give no additional details on the specific procedure we follow since is not important for all the subsequent operations. Any suitable technique to extract a set of circular dots from a generic scene would be fine.

At this point, we need a method to cluster all the candidate dots into different possible tags and discard all the erroneous ones that are originated by noise or clutter in the scene. Since we arranged the dots into circular rings, we expect that dots belonging to the same level would appear disposed around an elliptical shape once observed through a central projection. Therefore, dots in the same level can be identified by fitting an ellipse through their 2D image coordinates and verifying the distance assuming this model.

A common approach would consist in the use of a RANSAC scheme that uses a set of 5 candidate dots to estimate the model (i.e. the ellipse) against which quantify the consensus of all the others. Unfortunately, since 5 points are needed to characterize an ellipse into the image plane, the use of RANSAC in a scenario dominated by false positives (even without clutter we expect the majority of dots to belong to different tag

Total ellipses	10	50	100	500
Full (RANSAC)	252	2118760	75287520	$> 10^{10}$
Proposed method	45	1225	4950	124750

Figure 4.2: Number of maximum steps required for ellipse testing.

or even level) would quickly lead to an intractable problem (See Table 4.2). A possible alternative could be the use of a specialized Hough Transform [204], but also this solution would not be effective since it suffers by the relative low number of samples and the high dimensionality of the parameter space.

What makes possible the detection of our tags in reasonable time is the observation that there exist a relation between the shape of a dot and the shape of the ring in which is contained. Specifically, they both appear as ellipses (since they originate from a projection of two circles) and the parameters of both curves depend on the relative angle between the camera and the plane in which they lie. Even if from a single conic is not possible to recover the full camera pose, there is still enough information to recover (up to a finite set of different choices) a rotation that transform that conic into a circle. This, combined with a known relation between the relative size of the dots and the rings, can give clues of the geometry of a level and so ease the clustering process.

In this section, we give a detailed description on how the recovering of such rotation is done assuming a known camera matrix. In many situations, the requirement of a calibrated camera is not particularly limiting. For instance, if our tags would be used as a coarse registration method for a structured-light scanner solution (we give examples of this in section 4.3), the camera would certainly be calibrated as implied by the reconstruction process. However, for the high accuracy exhibited in points localization, it would be interesting to use our tags as a calibration target instead of a classical chessboard. To deal with this situations, we propose a way to still use our tags in an uncalibrated scenario in section 4.2.2.

Given the set of initial dot candidates, we start by recovering the parameters describing their elliptical shape. Specifically, we translate the image reference frame so that the principal point coincides with the origin, and parametrize each conic as the locus of point such that:

$$\vec{x}^T \mathbf{Q} \vec{x} = (u \quad v \quad 1) \begin{pmatrix} A & B & -\frac{D}{f} \\ B & C & -\frac{E}{f} \\ -\frac{D}{f} & -\frac{E}{f} & -\frac{F}{f^2} \end{pmatrix} \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = 0 \quad (4.1)$$

Where f is the camera focal length and u, v are pixel coordinates with respect to the optical center.

We follow [50] to estimate a rotation around the camera center that transforms the ellipse described by \mathbf{Q} into a circle. Specifically we decompose \mathbf{Q} via SVD

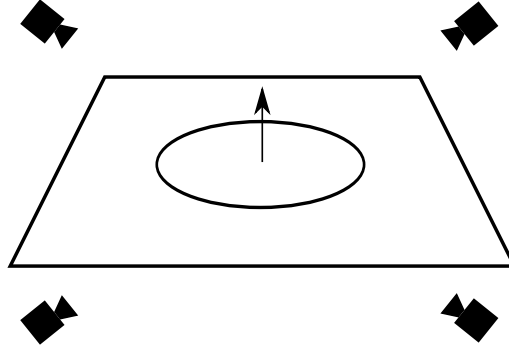


Figure 4.3: The four possible camera orientations that transform an observed ellipse into a circle

$$\mathbf{Q} = \mathbf{V}\Lambda\mathbf{V}^T \text{ with } \Lambda = \text{diag}(\lambda_1, \lambda_2, \lambda_3)$$

and compute the required rotation as:

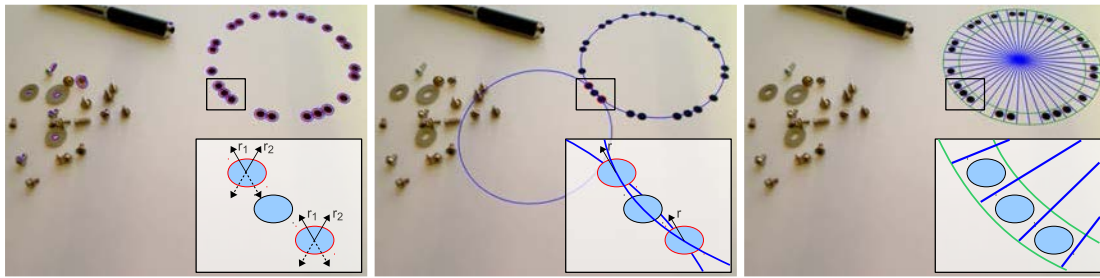
$$\mathbf{R}_Q = \mathbf{V} \begin{pmatrix} g \cos \alpha & s_1 g \sin \alpha & s_2 h \\ \sin \alpha & -s_1 \cos \alpha & 0 \\ s_1 s_2 h \cos \alpha & s_2 h \sin \alpha & -s_1 g \end{pmatrix} \quad (4.2)$$

$$g = \sqrt{\frac{\lambda_2 - \lambda_3}{\lambda_1 - \lambda_3}}, \quad h = \sqrt{\frac{\lambda_1 - \lambda_2}{\lambda_1 - \lambda_3}}$$

where s_1 and s_2 are two free signs, leaving 4 possible matrices, and α is any arbitrary rotation around the normal of the plane which remains constrained as long as we are observing just a single ellipse. At this point, if we fix $\alpha = 0$, each detected ellipse \mathbf{Q} may spawn four different rotation matrices $\mathbf{R}_Q^i, i = 1 \dots 4$ that transforms the conic into a circle (Fig. 4.3).

Since two of this four candidates imply a camera observing the back-side of the marker, we can safely discard all the \mathbf{R}_Q^i for which the plane normal $N_Q^i = \mathbf{R}_Q^i (0 \ 0 \ 1)^T$ is facing away from the camera (i.e. the last component is positive).

At this point, we search for whole markers by simultaneously observing the rotation matrices of couple of ellipses. Specifically, for each pair \mathbf{Q}_k and \mathbf{Q}_w , we produce the set of the four possible rotation pairs $\mathfrak{R} = \{(\mathbf{R}_{Q_k}^i, \mathbf{R}_{Q_w}^j); i, j = 1 \dots 2\}$. From this set, we remove the pairs for which the inner product of the relative plane normals is below a fixed threshold and average the remaining rotation pairs by means of quaternion mean. Finally, we keep the best rotation average by choosing the one that minimize the difference between the radii of \mathbf{Q}_k and \mathbf{Q}_w after being transformed by such rotation. The rationale is to avoid to choose ellipses with discordant orientations (as the marker



(a) Estimation of the feasible plane orientations

(b) Candidate ring estimation

(c) Dot vote counting

Figure 4.4: Steps of the ring detection: in (a) the feasible view directions are evaluated for each ellipse (with complexity $O(n)$), in (b) for each compatible pair of ellipses the feasible rings are estimated (with complexity $O(n^2)$), in (c) the dot votes are counted, the code is recovered and the best candidate ring is accepted (figure best viewed in color).

is planar) and to use a compatibility score that takes advantage of the fact that ellipses on the same ring should be exactly the same size on the rectified plane.

Whenever a pair of dots \mathbf{Q}_k and \mathbf{Q}_w generate a good average rotation $\mathbf{R}_{(\mathbf{Q}_k, \mathbf{Q}_w)}$, two hypothesis on the ring geometry can be made (Fig. 4.4.b). Indeed, we expect the ring shape being such that the following two properties holds. First, it should pass trough the centers of \mathbf{Q}_k and \mathbf{Q}_w . Second, the ratio between the ring radius and the radii of \mathbf{Q}_k and \mathbf{Q}_w , after being transformed trough $\mathbf{R}_{(\mathbf{Q}_k, \mathbf{Q}_w)}$, should be exactly κ . Operatively, we first fit the two circles $\mathbf{C}_1, \mathbf{C}_2$ passing trough the centers of $\mathbf{R}_{(\mathbf{Q}_k, \mathbf{Q}_w)}^T \mathbf{Q}_w \mathbf{R}_{(\mathbf{Q}_k, \mathbf{Q}_w)}$ and $\mathbf{R}_{(\mathbf{Q}_k, \mathbf{Q}_w)}^T \mathbf{Q}_k \mathbf{R}_{(\mathbf{Q}_k, \mathbf{Q}_w)}$ and having radius $\kappa \hat{r}$, where \hat{r} is the average radius of the two transformed dots. Then, we transform \mathbf{C}_1 and \mathbf{C}_2 back trough the inverse of $\mathbf{R}_{(\mathbf{Q}_k, \mathbf{Q}_w)}$.

As soon as candidate rings are extracted, a circular grid made by sector and levels can be generated directly on the image (Fig. 4.4.c). Of course, if the tag is composed by more than one level, we need to generate additional rings bot inward and outward. Then, for each slot the presence or absence of a dot can be observed to produce a binary sequence that will be analyzed in the decoding step to identify or discard the candidate marker.

To summarize, the detection step goal is to identify possible markers candidates by searching groups of dots belonging to the same ring, expecting them arranged in an elliptical shape. To do so, we avoid the direct estimation of ellipses in the image since it would require an unfeasible effort. Diversely, we take advantage of the geometrical properties of the dots and the known ratio κ to obtain two possible ring candidate for each pair of ellipses. As result, only $O(n^2)$ operations are required.

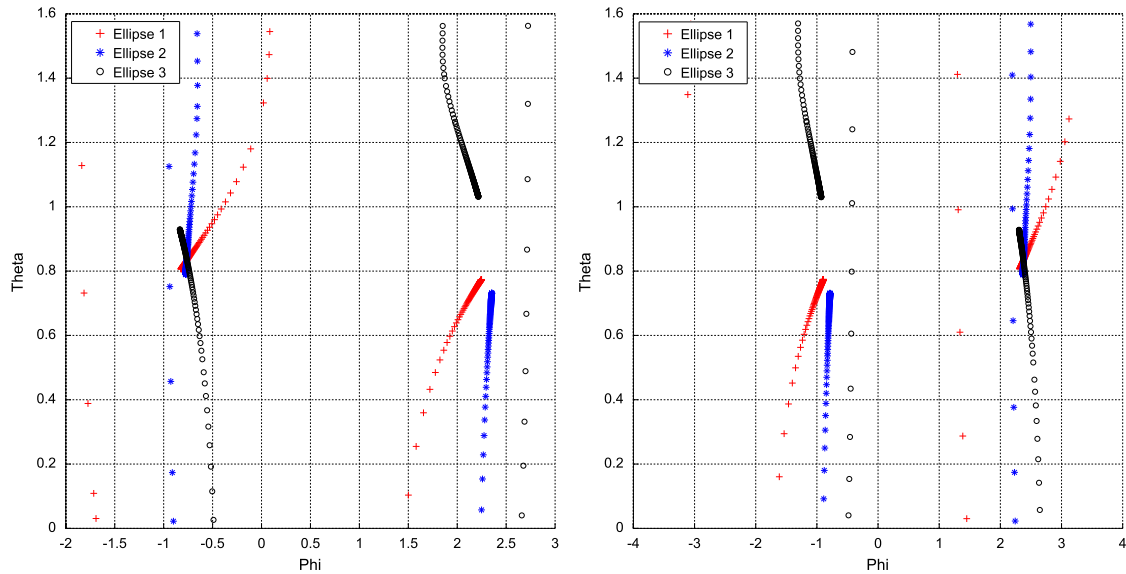


Figure 4.5: Estimated normals orientation in spherical coordinates of three coplanar ellipses spanning positive (Left) and negative (Right) focal length values. Note how one of the two possible orientations converge to a common direction while the other does the opposite.

4.2.2 Dealing with the uncalibrated case

The approach described so far assumed a calibrated camera setup. Indeed, all the rotation matrices \mathbf{R}_Q were designed to transform conics lying on the normalized image plane (hence requiring the focal length) around the camera optical center. It has to be noted, however, that camera intrinsics are not an implied requirement of the tag itself but just a powerful tool to dramatically speed-up the detection. As a consequence, we would be satisfied to just guess a raw estimation of the focal length and principal point good enough to still produce rotation matrices able to sustain the detection procedure.

We decided to use the image center as our guess of the principal point. Even if it appears a bold assumption, we observed that this holds for most cameras. Diversely, the focal length is difficult to guess as it depends on the lens mounted. However, also in this case we can take advantage on the geometric properties involved when observing a set of coplanar circles.

In Section 4.2.1 we discussed how two feasible plane normals can be estimated from each conic. It's crucial to observe that, if we apply the same projective transformation to two circles lying on the same plane, only one plane normal estimated from the first circle will be parallel to a normal extracted from a second, whereas the other two will diverge [50]. Furthermore, this property holds only for the correct focal length and optical center and can be naturally expanded to multiple coplanar conics.

To better explain this behaviour, we extracted all orientations from a set of 3 copla-

nar circles assuming to know the optical center and varying the focal length. In fig. 4.5 (Left) we plotted the values of such orientations in spherical coordinates spanning positive values of f from almost zero to 5 times the known correct value. For the right plot we did the same procedure but with negative values. In general, each ellipse produces two different traces in (ϕ, θ) -plane as a function of the focal length. Since all correct orientations have to be parallel to each other when the correct focal length is used, traces that are relative to the correct orientation will converge to a same point as f get closer to the expected value. On the other hand, all other traces will follow different paths and will diverge to different directions. It's clear from the plot that for positive values of the focal length there is only one intersection point (in this example $\phi \simeq -0.76, \theta \simeq 0.83$). Also, since the other possible intersection only happens when f becomes negative, the wrong orientation will never be present in the set of feasible solutions.

This means that we can both estimate the correct focal length and extract sets of coplanar circles by solving a clustering problem among all the generated plane normals. However, there is no simple closed form solution to reverse the process and obtain the best possible focal length that would have produced a given normal. Therefore, we restrict our estimation to a discrete set of n_f possible focal length values $f_i, i = 1 \dots n_f$ equally spaced inside the range $f_{\min} \dots f_{\max}$. At this point, for each dot \mathbf{Q} detected in a scene and for each f_i , exactly two feasible plane normals $N_{\mathbf{Q}_i}^1, N_{\mathbf{Q}_i}^2$ can be computed as described in section 4.2.1. All such normals will exhibit two degrees of freedom and hence can be easily parametrized in spherical coordinates with azimuth ϕ and elevation θ as vectors in \mathbb{R}^2 . Then, all these vectors are collected in a 2D accumulator whose bins are divided into equal angular ranges.

Once the accumulator is completely filled with values extracted from all the dots, local maxima with respect of the cardinality of the bins will represent clusters of normals oriented almost in the same direction¹. Finally, once a maximum is selected, we take the median focal length of all the candidates contained in a bin as our sought focal length estimate. Moreover, the candidates contained in a winning bin are all coplanar and thus the dots search phase can be restricted on such set.

An example of the proposed focal length estimation method is given in Fig. 4.6. We started by rendering a synthetic image of a RUNE-149 tag trough a virtual camera of known focal length $f_{vc} = 1000 \text{ px}$ and with principal point being exactly the center of the image (First row of Fig. 4.6). In the middle of the figure, we plotted the accumulator values projected on the front and back side of a hemisphere. As expected, a clear accumulation of votes can be observed in the bin containing the combination of ϕ, θ corresponding to the normal of the plane on which the tag lie. On the right, we plotted the distribution of the focal length candidates of the winning bin, highlighting a clear maximum around the correct value of f_{vc} . Conversely, we repeated the same test with two tags on the same image lying into two different planes (Second row of Fig. 4.6). This time, the accumulator shows two clear maxima corresponding to the plane nor-

¹more precisely, the variability inside the bin depends on the bin size itself

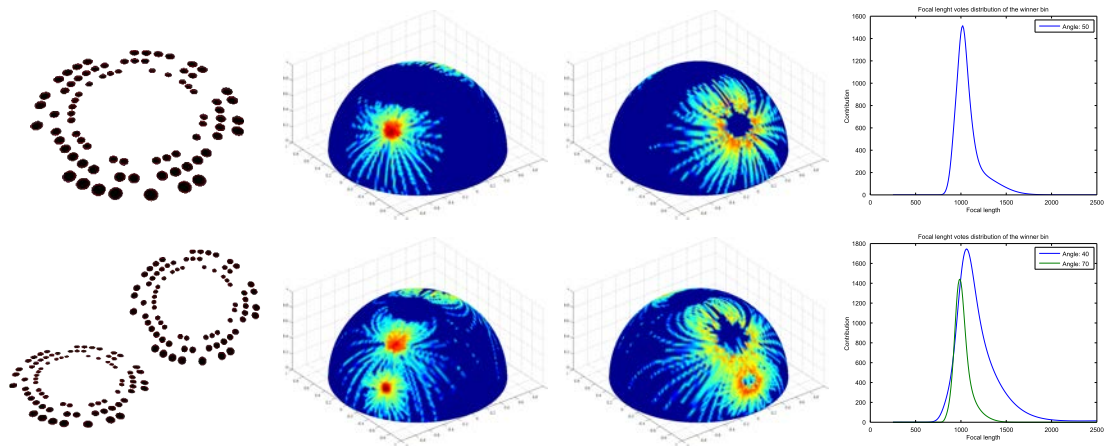


Figure 4.6: A synthetic representation of the marker dots normal voting scheme used to guess an initial value of the camera focal length. Left: RUNE-129 markers rendered by a virtual camera with known focal length and principal point. Center: the normal accumulator in spherical coordinates. Right: Focal length distribution of the bins. See the text for a complete discussion on the voting procedure.

mals of the two planes. Again, on the right side of the figure we plotted the distribution of the focal length candidates for both the two winning bins. Two important observations can be made. First, both the two distributions show two clear maxima around f_{vc} , demonstrating that a focal length guess is the same regardless of the tag orientation. Second, the more a tag is angled the more the guess is close to the expected value. This can be explained by noting that a tag perfectly parallel to the imaging plane has all the dots appearing as circles and so no focal length can be recovered. Therefore, the correct focal length is better constrained when the eccentricity of the dots is low. In fact, from the accumulator can be noted that the maximum corresponding to the angled tag is far more sharp than the other.

Even if the focal length guess is somehow biased by the angle of the observed tag, we feel that this won't be a show-stopper as we can still obtain a focal length guess good enough to let the detection procedure work properly. To convince the reader furthermore, we recall that the focal length is used to obtain a good rotation matrix to transform all the dots into circles. The more the angle is low, the more the focal length become irrelevant to recover that rotation. In the extreme case, to detect a perfectly parallel tag the focal length is not necessary at all since all the dots (and so the whole tag) already appear as circles.

To conclude, in the uncalibrated case we require an initial camera intrinsic parameters guessing step able to produce values good enough to perform a subsequent tag detection. To do so, we guess the principal point as the image center and the focal length with a voting procedure among a discretized set of plausible focal length values.

4.2.3 Marker Recognition and Coding Strategies

Once a candidate marker has been detected, dots distribution among the slots produces a sequence of symbols that can be subsequently used to identify each tag. However, two coupled problems raise. First, we don't have a starting position of the symbols sequence since the detection step can only identify each candidate up to a rotation around the normal of the plane². Consequently, any cyclic shift of the sequence is equally possible and must be recovered. Second, some dots may be missing or assigned to wrong slots thus requiring the identification procedure being somehow robust to this situations.

We decided to cast the problem into the solid mathematical framework of coding theory. Specifically, dot patterns of the tags corresponds to codes generated with specific properties and error-correcting capabilities. In section 4.2.3 we briefly discuss the mathematical theory involved in the generation of the codes while in section 4.2.3 we give a closed form solution to decode each code block in case of erasures and errors. We refer the reader to [120] for a in-depth investigation of the field.

Code generation

We start by defining a *block code* of length n over a set of symbols S as the set $C \subset S^n$. The elements of C are called *codewords*.

Let $q = p^k \in \mathbb{N}$ be a power of a prime number p and an integer $k > 1$. We denote with \mathbb{F}_q the finite field with q elements. A *linear code* C is a k -dimensional vector subspace of $(\mathbb{F}_q)^n$ where the symbols are taken over the field \mathbb{F}_q . A linear code is called *cyclic* if any cyclic shift of a codeword is still a codeword, i.e.

$$(c_0, \dots, c_{n-1}) \in C \Rightarrow (c_{n-1}, c_0, \dots, c_{n-2}) \in C$$

If we consider the field $\mathbb{F}_q[x]/(x^n - 1)$ obtained by the polynomial ring $\mathbb{F}_q[x]$ modulo division by $x^n - 1$, there exists a bijection to the vectors in $(\mathbb{F}_q)^n$:

$$(v_0, \dots, v_{n-1}) \Leftrightarrow v_0 + v_1x + \dots + v_{n-1}x^{n-1}$$

Furthermore, C is a cyclic code if and only if C is an ideal of the quotient group of $\mathbb{F}_q[x]/(x^n - 1)$. This means that all cyclic codes in polynomial form are multiples of a monic *generator polynomial* $g(x)$ of degree $m < n$ which divides $x^n - 1$ in $\mathbb{F}_q[x]$. Since multiplying a polynomial form of a code by x modulo $x^n - 1$ corresponds to a cyclic shift

$$x(v_0 + v_1x + \dots + v_{n-1}x^{n-1}) \bmod (x^n - 1) = v_{n-1} + v_0x + \dots + v_{n-2}x^{n-2} \quad (4.3)$$

²Note that, conversely, the verse of the sequence is induced by the counter-clockwise ordering of the sectors that is preserved since we always observe the frontal face of the marker plane.

all codewords can be obtained by mapping any polynomial $p(x) \in \mathbb{F}_q[x]$ of degree almost $n - m - 1$ into $p(x)g(x) \bmod (x^n - 1)$.

Since all the cyclic shift are codes, we can group the codewords into *cyclic equivalence classes* such that two codewords are in the same class if and only if one can be obtained as a cyclic shift of the other. Since the number of elements in a cyclic equivalence class divides n , by choosing an n prime we only have classes either composed by a single element (constant codewords with n repetitions of the same symbol) or where all codewords are distinct. The first can be easily eliminated since it involves in at most q codewords.

In our marker setting, the identity of the marker is encoded by the cyclic equivalence class while the actual alignment of the circles (i.e. its rotation around the plane normal) can be obtained from the detected element within the class. Using coding theory enable us to balance the trade-off between the number of errors that can be handled with respect to the number of possible valid tags (i.e. the number of equivalence classes) granted. This, to our knowledge, is the first fiducial marker system that provides such feature.

The *Hamming distance* $dH : S \times S \rightarrow \mathbb{N}$ is the number of symbols that differ between two codewords. Similarly, the hamming distance of a code C is the minimum distance between all the codewords: $dH(C) = \min_{u,v \in C} dH(u,v)$. The hamming distance plays a crucial role on the number of errors that can be detected and corrected. Indeed, a code with a hamming distance d can detect $d - 1$ errors and correct $\lfloor (d - 1) / 2 \rfloor$ erasures. When we consider a linear code of length n and dimension k , the singleton bound $d \leq n - k - 1$ holds. Thus, with a fixed code length n the error correcting codes capabilities are payed with a smallest number of available codewords. In our setting we restrict our analysis to the correction of random errors or erasures but the same mathematical framework can be used to improve the detection resilience while correcting burst errors (i.e. errors that are spatially coherent, like we have in case of occlusions).

For the proposed RENE-Tags, we experiment on two specific codes instances. In the first one (RUNE-43) we encode the single-layer circular pattern as a vector in $(\mathbb{Z}_2)^{43}$, where \mathbb{Z}_2 is the remainder class modulo 2. For this code we use the generator polynomial (4.4) which provides a cyclic code of dimension 15, 762 different markers (equivalence classes) with a minimum hamming distance of 13, allowing us to correct up to 6 errors.

$$g(x) = (1 + x^2 + x^4 + x^7 + x^{10} + x^{12} + x^{14})(1 + x + x^3 + x^7 + x^{11} + x^{13} + x^{14}) \quad (4.4)$$

In the second (RUNE-129) we have 8 different patterns (since is a 3-layers tag) in a sequence of 43 sectors. We hold out the pattern with no dots to detect erasures due to occlusions and we encode the remaining 7 patterns as vectors in \mathbb{Z}_7 . For the whole target, the code is represented as vectors in $(\mathbb{Z}_7)^{43}$ using the generator polynomial (4.5).

$$\begin{aligned}
g(x) = & (1 + 4x + x^2 + 6x^3 + x^4 + 4x^5 + x^6)(1 + 2x^2 + 2x^3 + 2x^4 + x^6) \\
& (1 + x + 3x^2 + 5x^3 + 3x^4 + x^5 + x^6)(1 + 5x + 5x^2 + 5x^4 + 5x^5 + x^6) \\
& (1 + 6x + 2x^3 + 6x^5 + x^6)(1 + 6x + 4x^2 + 3x^3 + 4x^4 + 6x^5 + x^6) \quad (4.5)
\end{aligned}$$

This provides a cyclic code of dimension 7 giving 19152 different markers with a minimum hamming distance of 30, allowing us to correct up to 14 errors, or 29 erasure, or any combination of e errors and c erasures such that $2e + c \leq 29$.

Decoding

The field \mathbb{F}_{76} splits $x^n - 1$ into n linear terms and there exists an $\alpha \in \mathbb{F}_{76}$ such that 28 consecutive powers of α ($\alpha^8 \dots \alpha^{36}$) are roots of the generator polynomial (4.5). Thus, for RUNE-129 we can use the BCH decoding algorithm [125, 58] to correct up to 14 errors or any combination of e errors and c erasures such that $2e + c \leq 28$.

Once we have decode the codeword, we can use integer Fourier Transform to find the cyclic shift of the observed code by setting zero the phase of the lowest non-zero frequency (AC) with non-zero amplitude.

While RUNE-43 is not a BCH code, a similar approach can be used to correct up to 5 errors. Note that, in both cases, our decoding procedure is not able to exploit the full error correcting capabilities of the code whereas we can still identify all the errors.

4.3 Experimental Validation

We tested our proposed fiducial markers in many different ways. To start, in section 4.3.1 and 4.3.2 we assessed the pose estimation accuracy compared to the ARToolkit[103] and ARToolkitPlus[185] which are deemed as a de-facto standard markers for augmented reality applications. Such tests are performed synthetically by rendering different frames varying an additive Gaussian noise, blur, illumination gradient and random occlusions.

Furthermore, driven by the good localization accuracy and occlusion resilience of the composing circular features, we tested RUNE-Tags as targets for camera calibration and object measurement. In section 4.3.3 we simulated a mono camera calibration scenario while in 4.3.4 we compared the camera pose estimation for both the mono and the stereo case. Also, we assessed the repeatability achievable while estimating the distance between two joint tags moving in a scene.

Finally, in addition to the evaluation with synthetic images, in section 4.3.6 we performed some qualitative tests on real videos.

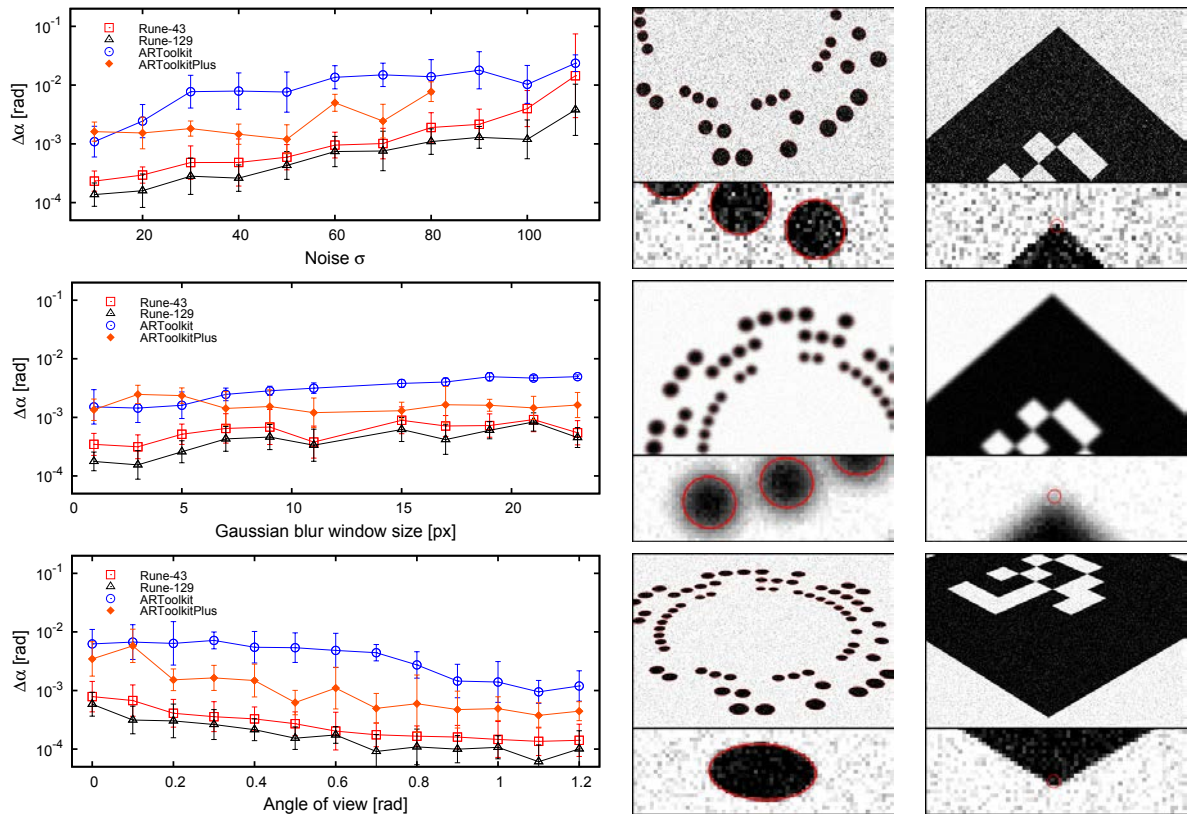


Figure 4.7: Evaluation of the accuracy in the camera pose estimation with respect to different scene conditions. Examples of the detected features are shown for RUNE-129 (first image column) and ARTToolkitPlus (second image column).

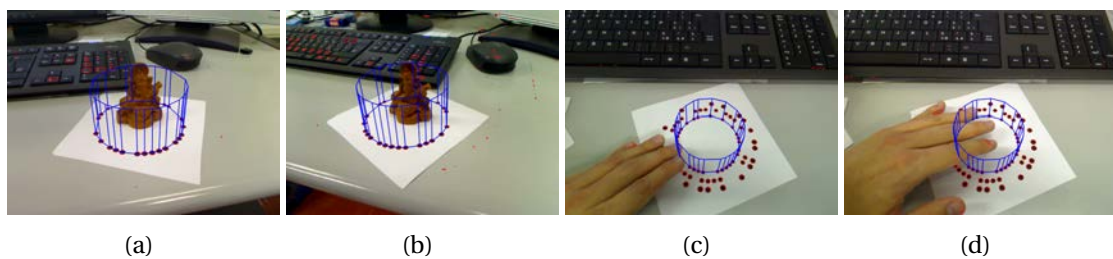


Figure 4.8: Some examples of behaviour in real videos with occlusion. In (a) and (b) an object is placed inside the marker and the setup is rotated. In (c) and (d) the pose is recovered after medium and severe occlusion.

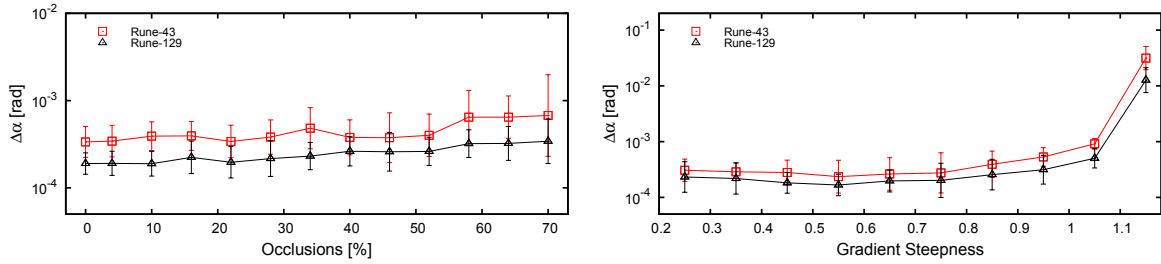


Figure 4.9: Evaluation of the accuracy in the camera pose estimation of RUNE-Tag with respect to occlusion (left column) and illumination gradient (right column).

4.3.1 Accuracy and Baseline Comparisons

In Fig. 4.7 the accuracy of our markers is evaluated. In the first test, an additive Gaussian noise was added to images with an average view angle of 0.3 radians and no artificial blur added. The performance of all methods get worse with increasing levels of noise and ARToolkitPlus, while in general more accurate than ARToolkit, breaks when dealing with a noise with a std. dev. greater than 80 (pixel intensities goes from 0 to 255). Both RUNE-43 and RUNE-129 always recover a more faithful pose. We think that this is mainly due to the larger number of correspondences used to solve the PnP problem. In fact, we can observe that in all the experiments RUNE-129 performs consistently better than RUNE-43. Unlike additive noise, Gaussian blur seems to have a more limited effect on all the techniques. This is mainly related to the fact that all of them performs a preliminary edge detection step, which in turn applies a convolution kernel. Thus is somewhat expected that an additional blur does not affect severely the marker localization. Finally, it is interesting to note that oblique angles lead to an higher accuracy (as long as the markers are still recognizable). This is explained by observing that the constraint of the reprojection increases with the angle of view. Overall, these experiments confirm that Rune-Tag always outperforms the other two tested techniques by about one order of magnitude. In practical terms the improvement is not negligible, in fact an error as low as 10^{-3} radians still produces a jitter of 1 millimeter when projected over a distance of 1 meter. While this is a reasonable performance for augmented reality applications, it can be unacceptable for obtaining precise contactless measures.

4.3.2 Resilience to Occlusion and Illumination

One of the main characteristics of Rune-Tag is that it is very robust to occlusion. In section 4.2.3 we observed that RUNE-129 can be used to distinguish between about 20.000 different tags and still be robust to occlusions as large as about 67% of the dots. By choosing different cyclic coding schemes is even possible to push this robustness even further, at the price of a lower number of available tags. In the first column of Fig. 4.9 we show how occlusion affects the accuracy of the pose estimation (i.e. how well the

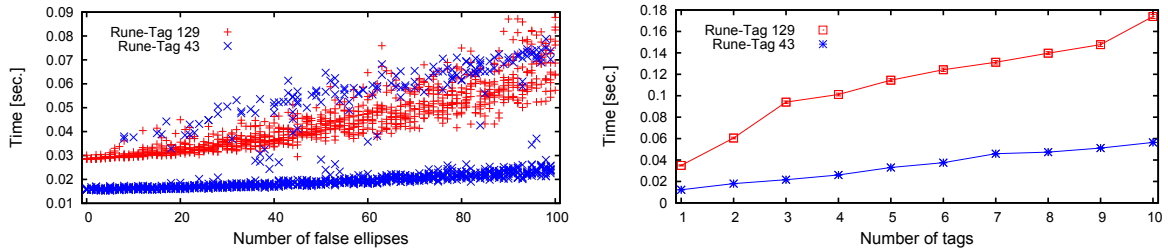


Figure 4.10: Evaluation of the recognition time respectively when adding artificial false ellipses in the scene (left column) and with several markers (right column).

Occlusion	0%	10%	20%	50%	70%
RUNE-43	100%	69%	40%	0%	0%
RUNE-129	100%	100%	100%	100%	67%

Figure 4.11: Recognition rate of the two proposed marker configurations with respect to the percentage of area occluded.

pose is estimated with fewer dots regardless to the ability of recognize the marker). Albeit a linear decreasing of the accuracy with respect to the occlusion can be observed, the precision is still quite reasonable also when most of the dots are not visible. In Fig. 4.11 we show the recognition rate of the two proposed designs with respect to the percentage of marker area occluded. In the second column of Fig. 4.9 the robustness to illumination gradient is examined. The gradient itself is measured unit per pixel (i.e. quantity to add to each pixel value for a each pixel of distance from the image center). Overall, the proposed methods are not affected very much by the illumination gradient and break only when it become very large (in our setup an illumination gradient of 1 implies that pixels are completely saturated at 255 pixels from the image center).

4.3.3 RUNE Tags for camera calibration

Since RUNE-129 provides an extremely robust yet precise way to localize many circular features we tried to use the proposed markers as a calibration target. In most cases, camera calibration is performed by exposing a well known pattern to the camera in many different point of views. This allows the gathering of many 3D-2D point correspondences used to simultaneously recover the target pose, camera intrinsics parameters, and the lens distortion. Most of the time, a chessboard pattern is used since it provides a good set of feature points in the form of image corners. However, a manual chessboard boundary identification process is required for the limited robustness of such patterns against occlusions or illumination gradients. As a consequence, our fiducial markers may provide a very interesting alternative when an automatic calibration procedure is sought or an optimal visibility of the target cannot be guaranteed.

In Fig.4.12 we show the calibration results while calibrating a camera using a single RUNE-129 as calibration target and by varying the number of exposures used for each calibration. Specifically, we divided the camera poses (as given by PnP) into 3 major groups with respect to the angle between the camera z -axis and the marker plane normal. For each group, more than 200 photos are taken and a random subset of them are selected for each test to compose the plot. The ground truth is provided by a calibration performed with a 20×20 chessboard target exposed in 200 different poses using the method described in [19] to limit the errors due to printing misalignments. Camera calibration is performed by using the common technique described in [206] implemented by the OpenCV library [36].

Some interesting facts can be observed. First, the standard deviation of all the estimated parameters decrease by increasing the number of photos. This is an expected behaviour that agrees with the accuracy experiments presented in section 4.3.1. Indeed, the more the number of target feature points given, the more the calibration error can be reduced by the non-linear optimization process. Second, the focal length estimation tends to be more accurate while considering the target poses spanning through a greater range of angles (i.e. between 0 and 60 degrees). Differently, optical center seems to behave in the opposite way, giving better results when keeping the target plane more parallel to the image plane. This is probably due to the well known localization bias of the ellipse centers [129]. Third, the first two radial distortion parameters (i.e. k_1 and k_2) behave respectively like the optical center and the focal length. It has to be noted that a precise localization of the ellipse centers is only achievable in absence of distortion since the projective invariance of conics holds only for pure central projections. Therefore, we think that the calibration performance can be improved by estimating an initial calibration assuming no radial distortion followed by an iterative undistortion and re-localization of the circular features and a re-calibration of the camera intrinsics. Finally, we obtained no completely wrong calibrations due to mis-detections of the target thanks to the extremely resilient coding scheme used for markers identification.

4.3.4 Mono vs. Stereo Pose Estimation

To further test the camera pose estimation accuracy we compared the results achievable comparing a single camera setup (using PnP algorithm) with a calibrated stereo setup that can recover the pose by means of a 3D reconstruction of the marker followed by an estimation of the rigid motion with respect to the known model.

We started by calibrating a stereo rig using a marker-based target as described in section 4.3.3. Then, we firmly positioned two RUNE-129 tags to a rigid metal rod so that they could only be moved without changing their relative position.

In the first experiment (Fig. 4.13, Left) we plotted the unknown distance between the two markers as estimated only by the first camera (in red), by the second (in green) and by using stereo reconstruction (blue) as a function of the angle between the first marker and the first camera. It can be noted that the stereo case exhibit lower vari-

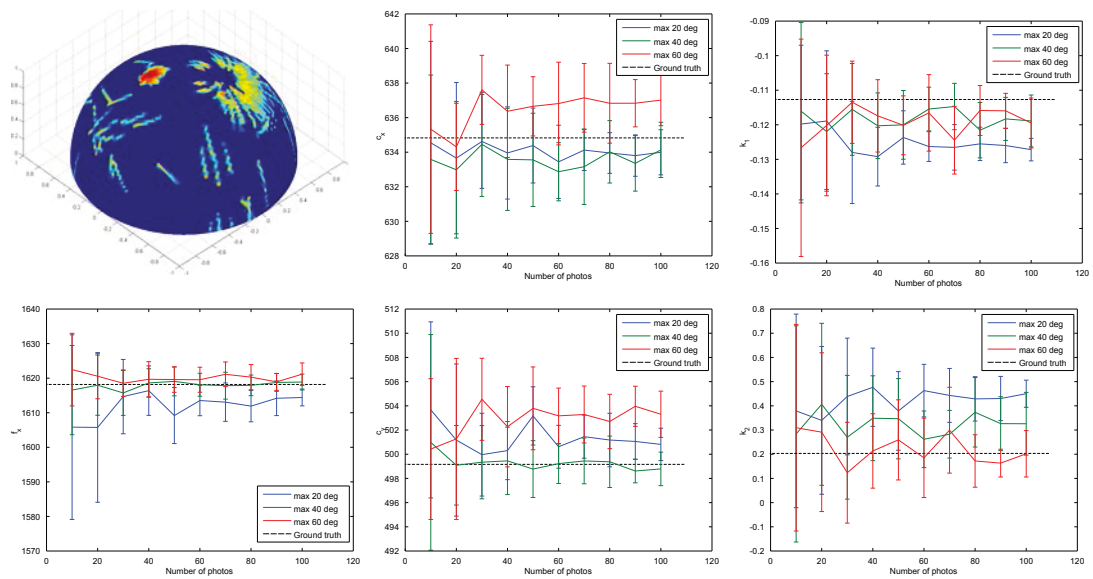


Figure 4.12: Accuracy of camera calibration while using a single RUNE-129 as a dot-based calibration target. Camera poses has been divided into 3 groups based on the maximum angle between the camera z -axis and the marker plane. A random subset of photos is used to test the calibration varying the number of target exposures. In all the experiments we achieve a good accuracy with a decreasing st.dev. when increasing the number of photos.

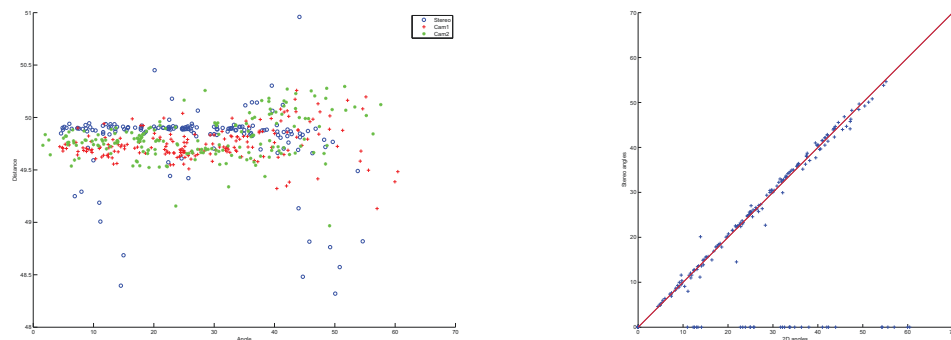


Figure 4.13: Comparison between the pose accuracy for a single or stereo camera setup. Left: distance between two jointly moving markers as a function of the angle with respect to the first camera. Right: Angle around the marker plane normal as estimated by the first camera versus the stereo setup. Ideally, all the measures should lie on the 45 degrees red line.

ance with respect to both only the first and the second camera. Moreover, the distance measured by the mono case tends to be a little lower than the stereo one if the angle is below 30 degrees while increasing significantly for higher angles. This behaviour is probably due to the PnP algorithm that suffers for a non-isotropic error with respect to the three camera axis (i.e. the localization error on the camera z -axis is higher than the other two).

In (Fig. 4.13, Right) we compared the angle around the plane normal of a single RUNE-129 tag for mono (using the first camera) versus the stereo case. Ideally, the ratio between the two measures should be exactly 1 and so all the points should be disposed on the 45 degrees red line shown in the plot. We can observe that most of the measures are equally distributed above and below such line indicating no significant bias. This behaviour is consistent for all the angles spanning between 10 and 60 degrees since the overall geometrical shape of all the dots (i.e. minor and major axis length) remains constant if a rotation around the marker plane normal is applied. This suggests that the proposed tags may be used as a coarse registration initialization for a 3D scanner turntable.

4.3.5 Performance Evaluation

Our tag system is designed for improved accuracy and robustness rather than for high detection speed. This is quite apparent in Fig. 4.10, where we can see that the recognition could require from a minimum of about 15 ms (RUNE-43 with one tag and no noise) to a maximum of about 180 ms (RUNE-129 with 10 tags). By comparison ARToolkitPlus is about an order of magnitude faster [185]. However, it should be noted that, despite being slower, the frame rates reachable by Rune-Tag (from 60 to about 10 fps) can still be deemed as usable even for real-time applications (in particular when few markers are viewed at the same time).

4.3.6 Shortcomings and Limitations

In Fig. 4.8 some experiments with common occlusion scenarios are presented. In the first two shots an object is placed inside a RUNE-43 marker in a typical setup used for image-based shape reconstruction. In the following two frames a RUNE-129 marker is tested for its robustness to moderate and severe occlusion. At last, in Fig. 4.14 an inherent shortcoming of our design is highlighted. The high density exhibited by the more packed markers may result in a failure of the ellipse detector whereas the tag is far away from the camera or very angled, causing the dots to become too small or blended.

4.4 Conclusions

In this chapter we described a novel fiducial marker system which heavily relies on the robust framework of cyclic codes to offer superior occlusion resilience, accurate

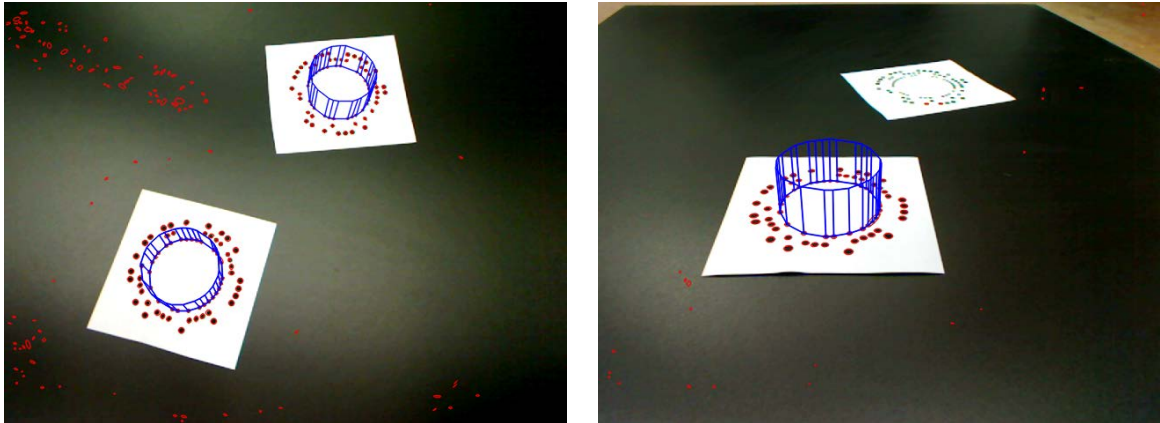


Figure 4.14: Recognition fails when the marker is angled and far away from the camera and the ellipses blends together.

detection and robustness against various types of noise. We improved on the seminal version proposed in [31] by investigating their usage even for the uncalibrated case and developed a fast technique to directly decode the symbol sequence encoded in each tag.

Moreover, we expanded the experimental evaluation by testing its adequacy to be used as a typical dot-based camera calibration target. This is supported by also experiments to evaluate its behaviour comparing the pose estimation with both the mono and the stereo scenario.

Overall, RUNE-Tags offer many advantages over the existing fiducial marker systems. First, it gives the user the flexibility to trade-off the occlusion resilience with respect to the number of simultaneously detectable tags in a scene. Consequently, if one favors robustness against diversity, a very limited set of tags can be generated with high hamming distance to guarantee extremely high error recovery rates. Second, the design itself may vary in the number of possible layers. The proposed single-layered RUNE-43 exhibit limited occlusion resilience while offering plenty of space in the marker interior for any additional payload or even for placing a physical object for reconstruction task. Third, by providing many circular features on a single tag we not only achieve an order of magnitude better pose recovery with respect to other tags but we managed to use the tag itself as a calibration target. So, we imagine RUNE-Tag being possibly used for measuring large distances or objects.

Finally, while slower than other techniques, our method is fast enough to be used in real-time applications. However, the severe packing of circular points may cause the ellipse detector to fail especially when dealing with low resolution, high angles or motion-blurred images. This limitation can be easily relieved by using a simpler marker, such as RUNE-43, which allows for a more extended range while still providing a satisfactory precision.

5

Camera Calibration from Coplanar Circles

In this chapter we exploit the projective properties of conics to estimate the focal length and optical center of a pinhole camera just by observing a set of coplanar circles, where neither the radius nor the reciprocal position of each circle has to be known a-priori. This makes such method particularly interesting whenever the usage of a calibration target is not a feasible option.

The contribution presented in this chapter is twofold. First, we propose a reliable method to locate coplanar circles from images by means of a non-cooperative evolutionary game. Second, we refine the estimation of camera parameters with a non-linear function minimization through a simple yet effective gradient descent. Performance of the proposed approach is assessed through an experimental section consisting on both quantitative and qualitative tests.

5.1 Introduction

The recovery of all parameters that determine the inner working of an imaging device is often a mandatory step to allow further analysis of a scene. For instance, 3D reconstruction, photogrammetry and in general every process that involves geometrical reasoning on objects deeply rely on such knowledge. For many applications, it is a good assumption to consider the camera to behave according to the pinhole model, thus requiring only the estimation of the focal length (the distance between the projection center and the image plane) and the optical center (coordinates of the intersection of the optical axis with the image plane). Moreover, if we are dealing with a generic camera, this model can be further improved by considering non-square pixels (thus discriminating between horizontal and vertical focal length), skewness (non rectangular pixels) or radial distortion modelled in a number of different ways.

If we aim at the best possible accuracy, it is universally considered a successful approach to rely on a known calibration target and exploit techniques that are able to recover camera parameters by exposing such object to the camera under different poses. A widely adopted technique is the one introduced in [206] based on a planar pattern composed by hundred of corner points for easy localization. Furthermore, Heikkilä [88] exploit a grid of circular points to accurately reverse the distortion model. An interesting variation on the theme is proposed in [19] with a technique that can simultaneously optimize camera and target geometry to overcome manufacturing imperfections of the target.

However, equally interesting are self-calibration methods that just rely on rigidity of the scene or known geometric primitives that are assumed to be present. Good candidates for such primitives are lines and conics since they are all invariant to any projective transformation. For example, [198] assumes a scene in which a circle and a coplanar point at infinity can be detected. In [135] the authors use a similar setting but with a pencil of lines passing through its center. For their interesting properties, circles are besides used in [57] and [50] assuming such primitives to be coaxial or coplanar.

Our work falls also in this category. Specifically, we exploit a scene in which we assume some coplanar circles are visible and relate the shape of such conics with the parameters of a zero-distortion pinhole camera. We believe that this setting is common (or anyway simple to setup) in many urban and indoor scenarios, so offering a good trade-off between estimation accuracy and method feasibility. Despite the aforementioned methods, that aim to depend on as few primitives as possible, we exploit all circular features available in a scene to minimize the estimation error.

This chapter is organized as follows. In Section 5.2 we propose a novel technique to extract a reliable cluster of coplanar circles from a general scene, together with an initial guess of the camera parameters. In section 5.3 we refine such guess by means of a gradient descent minimization of an energy functional that accounts for the eccentricity of ellipses undergoing a central projection. Finally, in section 5.4 we show some experimental results obtained with different synthetically generated images together with a qualitative evaluation of its behaviour when applied to real world scenarios.

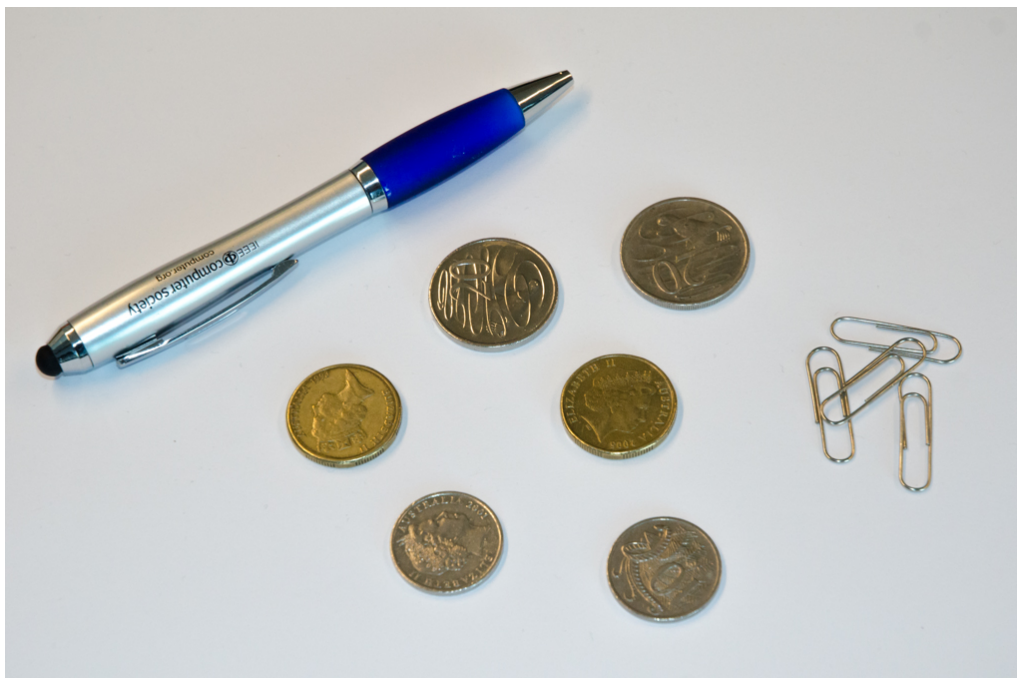


Figure 5.1: A possible calibration target composed by some coplanar circles.

5.1.1 Our approach

In this work we generalize the concepts previously introduced in the uncalibrated case of RENE-Tag detection (Section 4.2.2) by taking into account multiple shots of a scene in which many coplanar circles can be seen. Not the camera parameters, nor the orientation for each pose are known. We just assume that the camera focal length and optical center do not change among the shots.

After detecting all the ellipses for each scene, we exploit the normal alignment property to obtain an initial guess of all the orientations and camera parameters. Differently from the accumulator approach presented before, we follow [18, 20] to cast the problem in a framework in which each possible camera orientation plays as a strategy in a sequence of non-cooperative games. The outcome of such process is twofold. First, ellipses originated from non-coplanar circles will generate uncorrelated orientations with the others and they will be likely be excluded from the winning strategies. As a result, a very reliable set of coplanar circles can be clustered from each shot. Second, since such orientations depend on the camera intrinsic parameters, we get an initial guess by selecting the parameters providing the best alignment among the selected orientations.

5.1.2 Ellipse detection and refinement

Our calibration method starts by extracting all possible ellipses from each scene. Many different methods have been proposed in literature, for instance in [194, 144] and [166]. To maintain the whole method simple, we just opted for a combination of image threshold, closed-contour extraction and ellipse fitting. To keep the detection independent from threshold parameters and ensure a precise sub-pixel fitting of the contour, we further refine the coefficients of each ellipse as described in [147]. In general, we don't require a specific detection procedure as long as special care is taken to use methods that are able to precisely estimate the size of the extracted ellipses.

5.2 Selecting Coplanar Circles with a Non Cooperative Game

After the ellipse detection step, we obtain a set of N ellipse matrices for each image. Hopefully, some of those describe coplanar circles actually present in the scene, while others just resulted from different visible curvilinear objects, mis-detections and so on. Since the subsequent optimization procedure cannot implicitly handle any outlier, it is crucial to select only a reliable set of trusted coplanar circles. Moreover, we need to give a rough guess of the initial focal length, optical center, and camera orientations.

For optical center, we just guess its initial value in pixel as the center of the image, i.e. half image width and height. Despite being a naive approximation, we found that it's a reasonable starting point for most of typical camera optics and we experimentally observed that the optimization is not very sensible to this parameter (See sec. 5.4).

Then, we proceed by discretizing a range of possible focal length values in k equal intervals. For example, if we expect that f may be contained in the interval $[f_{\min} \dots f_{\max}]$ we create a set of k candidate focal length values $f_i = f_{\max} \frac{i}{k-1} + f_{\min} (1 - \frac{i}{k-1})$, $i = 0 \dots k-1$. For each candidate focal length f_i , a set of $2N$ possible orientations $O_s = o_1 \dots o_{2N}$ can be computed from all the ellipses, as described before. These orientations are vectors in \mathbb{R}^3 representing the plane normal in which each ellipse lie in the 3D-space.

We then define a payoff function between pairs of orientations as:

$$\pi(o_i, o_j) = \langle o_i, o_j \rangle$$

where $\langle \cdot, \cdot \rangle$ is the standard vector inner product. We further arrange all possible payoff values in a symmetric square matrix

$$\Pi \in \mathbb{R}_{(2N \times 2N)}, \Pi_{i,j} = \pi(o_i, o_j)$$

Since we expect all the correct orientations be parallel to each other, we aim to extract the largest possible subset of O_s for which π is maximized between all the couples.

To select the compatible orientations we adopt the Game-Theoretic matching and selection process presented in [18, 154]. We run a different game for each candidate fo-

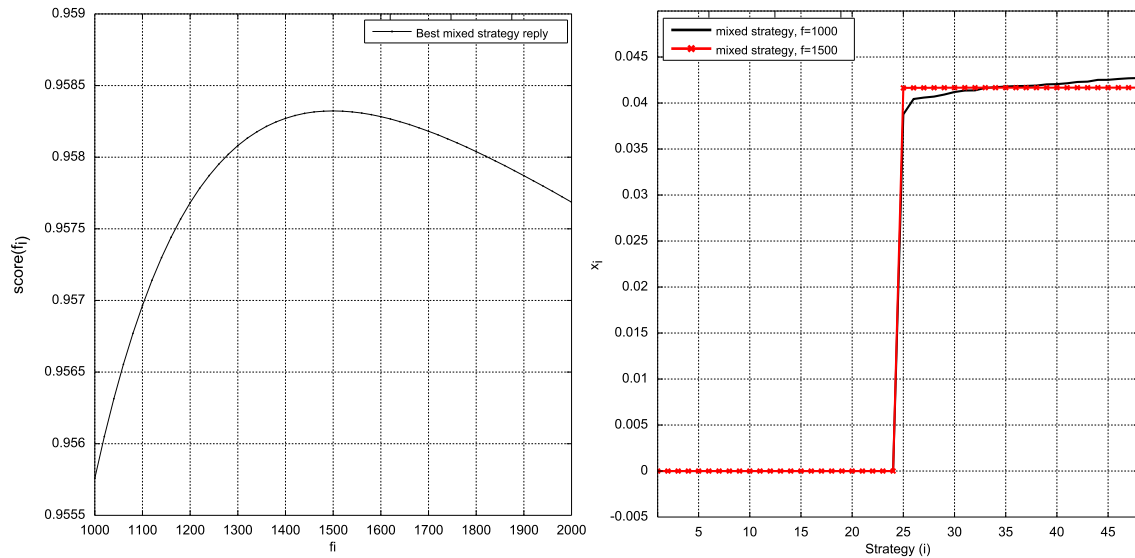


Figure 5.2: Left: Score obtained by a set of games played for different candidate focal length spanning around the correct known value of 1500. Note the clear maximum obtained around such value. Right: Final population after two non-cooperative games with the correct focal length value ($f=1500$) and a wrong value ($f=1000$). In the correct case, almost all winning strategies will exhibit the same value.

cal length f_i , and select the focal length, the orientation and the set of coplanar circles from the game with the maximum average support

$$score(f_i) = \vec{x}^T \Pi \vec{x}$$

where \vec{x} the stable mixed strategy selected by the evolutionary process.

In fig. 5.2 (left) we plotted an instance of such score for a set of f_i candidates spanning from 1000 to 2000 around the ground truth correct value of 1500. The curve originated by the function appears smooth with a clear maximum on the desired value. Moreover, it's interesting to observe that the more f_i gets closer to the real value, the more the winning strategies in \vec{x} will exhibit the same contribution since all equally important for the equilibrium of the game. An example of this behaviour is shown in fig. 5.2 (right).

5.3 Camera parameters optimization

After clustering coplanar circles and guessing initial camera parameters, a fine estimation is performed by optimizing an energy function that accounts for the eccentricity variation of the ellipses with respect to such parameters.

5.3.1 Problem formulation

Let $\mathbf{E}_1^m \dots \mathbf{E}_{N(m)}^m$ be the 3×3 matrices describing the $N(m)$ ellipses representing each coplanar circle found in the image m after the game-theoretic clustering. Let f' , c'_x and c'_y be the initial camera intrinsic guesses, and (α_i, α_{i+1}) , $i = 1 \dots 2M$ be the euler angles with respect to x and y axis for camera orientation in the image $\text{floor}((i-1)/2 + 1)$.

Moreover, let:

$$\begin{aligned}\vec{\alpha} &= (\alpha_1, \alpha_2, \dots, \alpha_{2M})^T \in \mathbb{R}^{2M} \\ \vec{x} &= (\phi_1, \phi_2, \phi_3) = \left(\frac{c'_x}{f'}, \frac{c'_y}{f'}, \frac{1}{f'} \right) \\ \mathbf{C}_f &= \begin{pmatrix} 1 & 0 & \phi_1 \\ 0 & 1 & \phi_2 \\ 0 & 0 & \phi_3 \end{pmatrix}\end{aligned}$$

Under the assumption that each \mathbf{E}_i^m is effectively representing circle \mathbf{Q}_i^m in the 3D world space, its equation can be expressed as a function of the camera parameters and rotations around the optical center through the following transformation:

$$\mathbf{Q}_i^m(\mathbf{R}_c, \vec{x}) = \mathbf{R}_c^T \mathbf{C}_f^T \mathbf{E}_i^m \mathbf{C}_f \mathbf{R}_c \quad (5.1)$$

where \mathbf{R}_c is the rotation matrix that lets the circles plane be parallel to the image plane. Since the transformation has the same effect up to any rotation around the axis $(0, 0, 1)^T$, we define:

$$\mathbf{A}_i^m(\vec{\alpha}, \vec{x}) = \mathbf{R}_m^T \mathbf{C}_f^T \mathbf{E}_i^m \mathbf{C}_f \mathbf{R}_m \quad (5.2)$$

$$\mathbf{R}_m = \begin{pmatrix} \cos(\alpha_{2m+1}) & 0 \\ \sin(\alpha_{2m}) \sin(\alpha_{2m+1}) & \cos(\alpha_{2m}) \\ -\cos(\alpha_{2m}) \sin(\alpha_{2m+1}) & \sin(\alpha_{2m}) \end{pmatrix}$$

If \mathbf{Q}_i^m describes a non singular circle, the two eigenvalues of $\mathbf{A}_i^m(\vec{\alpha}, \vec{x})$ must be real and equal. For each ellipse i in the image m , the two eigenvalues of $\mathbf{A}_i^m(\vec{\alpha}, \vec{x})$ can be easily computed in a closed form as:

$$\lambda_{1,2} = \frac{\text{tr}(\mathbf{A}) \pm \sqrt{\text{tr}(\mathbf{A})^2 - 4\det(\mathbf{A})}}{2}$$

Moreover, we define $\text{gap}(\mathbf{A}_i^m(\vec{\alpha}, \vec{x}))$ as the square of the distance between the two eigenvalues:

$$\text{gap}(\mathbf{A}_i^m(\vec{\alpha}, \vec{x})) = \text{tr}(\mathbf{A})^2 - 4\det(\mathbf{A}) \quad (5.3)$$

Given that, for each possible real matrix \mathbf{M} , it holds that

$$\text{tr}(\mathbf{M}) = \log(\det(\exp(\mathbf{M})))$$

and so the determinant can also be expressed as

$$\det(\mathbf{M}) = \frac{\text{tr}(\mathbf{M})^2 - \text{tr}(\mathbf{M}^2)}{2}$$

Hence, we can re-write the equation (5.3) as:

$$\text{gap}(\mathbf{A}_i^m(\vec{\alpha}, \vec{x})) = 2\text{tr}(\mathbf{A}_i^m(\vec{\alpha}, \vec{x})^2) - \text{tr}(\mathbf{A}_i^m(\vec{\alpha}, \vec{x}))^2 \quad (5.4)$$

5.3.2 Energy minimization

Let

$$\text{Energy}(\vec{\alpha}, \vec{x}) = \sum_{m=0}^{M-1} \sum_{i=1}^{N(M)} \text{gap}(\mathbf{A}_i^m(\vec{\alpha}, \vec{x})) \quad (5.5)$$

It is possible to estimate the focal length f , the optical center (c_x, c_y) and the rotation angles with respect to $(1, 0, 0)^T$ and $(0, 1, 0)^T$ axes by solving the following minimization problem:

$$\underset{\vec{\alpha}, \vec{x}}{\text{argmin}} \text{Energy}(\vec{\alpha}, \vec{x}) \quad (5.6)$$

Indeed, the energy functional is minimized when each matrix $\mathbf{A}_i^m(\vec{\alpha}, \vec{x})$ has two equal eigenvalues, namely when $\text{gap}(\mathbf{A}_i^m(\vec{\alpha}, \vec{x})) = 0$.

We implemented a gradient descent scheme to optimize the given functional starting from the guesses obtained from the game-theoretic step. Moreover, no approximation of the function partial derivatives is needed since $\nabla \text{Energy}(\vec{\alpha}, \vec{x})$ can be easily obtained in a closed form with any symbolic math tool.

5.4 Experimental Evaluation

We decided to give more emphasis to experiments performed on synthetically generated images since the ground truth is known and we can better highlight various aspects of the approach. Qualitative experiments were also performed on real world data to demonstrate its feasibility in different possible scenarios.

Synthetic Data

We rendered synthetic scenes composed by coplanar circles with random radius and position in images 800 pixels wide and 600 pixels high. We simulated a pinhole camera with no distortion, a focal length of 1500 px and optical center $c = (400, 300)$. Rasterization was performed using Qt libraries with anti-aliasing hinting enabled.

We started by testing the sensitivity of the refinement step to the initial intrinsic guess. This is an important assessment because, even if the inlier selector does a great job in clustering the ellipses, it still depends on how we discretize the focal length search domain. A coarse discretization of a very broad range will probably cause a guess quite far from the correct one. In fig. 5.4 we plotted the absolute error obtained for the focal length as a function of the distance between the correct and the guessed focal length while in fig. 5.5 we did the same varying the optical center. From the experiment, there appear no evident correlation between them, proving that the optimization is still able to converge close to the correct solution while not suffering so much for local minima.

In fig. 5.6 we assessed the calibration accuracy varying the number of poses and inserting additive gaussian noise to the image. Specifically, in the first row we plotted the obtained average and standard deviation of focal length and optical center while calibrating 7 different sets of 24 coplanar circles varying the number of poses. No particular improvement is observable giving almost the same results ranging from 6 to 27 different poses. This is somehow expected since it's in theory possible to calibrate with just one shot. In the second row we have perturbed each image pixel with zero-mean gaussian noise and given sigma. Also in this case, we repeated the experiment many times with different set of circles. The obtained result do not deviate much from the expected value but we can observe a variance increase as the noise become more and more effective.

From all the tests performed emerge a little calibration bias due to the high sensitivity of the method to the size of the ellipses. Since those sizes depend on image blurring and is not isotropic over the circle plane as is not parallel to the camera image plane, we feel that great care have to be taken while detecting and fitting ellipses on the image. As a result, for any application requiring very small calibration error, the usage of a precise well-crafted known target is unavoidable. However, the proposed method offers a good balance between the accuracy (relative error obtained in our tests was always less than 1%) and flexibility offered by the fact that requires no previous knowledge of the scene.

Real-world scenarios

Our proposed calibration pipeline was also tested with some real world pictures freely accessible from the web. Since ground truth was not available, we just rectified such images from the estimated parameters to give a qualitative overview of the achievable results. We point out that no particular care was used to tune the ellipse detection stage. Still, the coplanar inlier selector was able to correctly identify a reliable cluster to perform the estimation. The input images used, along with the computed rectification, are shown in fig. 5.3.

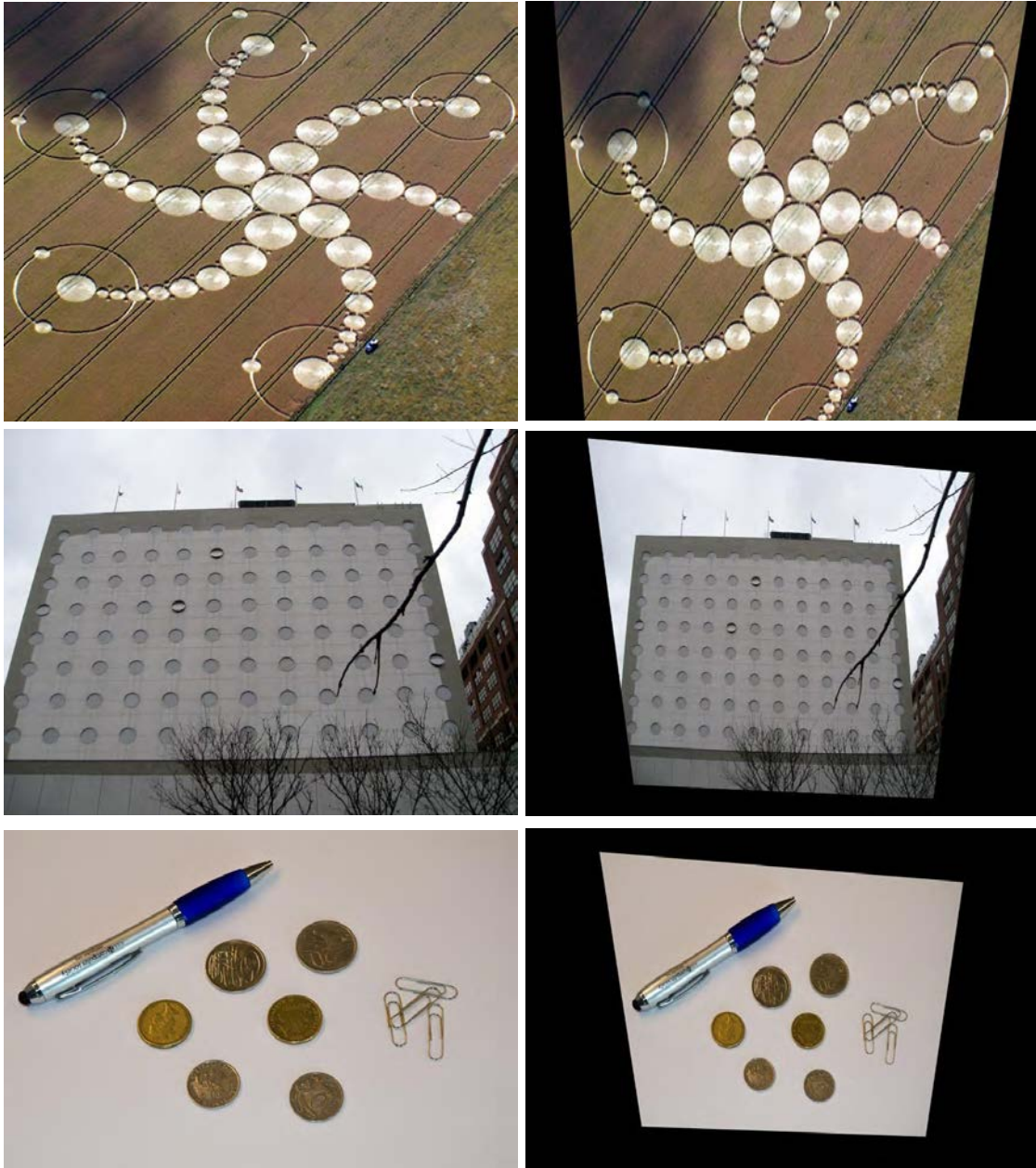


Figure 5.3: Some qualitative calibration examples on real world scenes. Left: Original images. Right: Rectified images obtained from the estimated camera intrinsics and orientation.

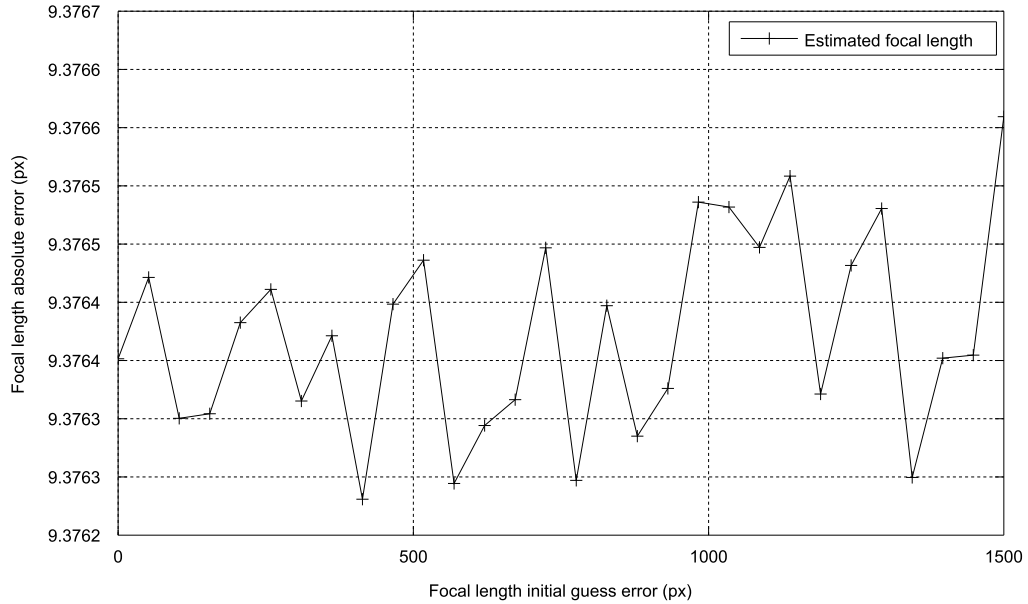


Figure 5.4: Focal length estimation error after the optimization varying the initial focal length guess.

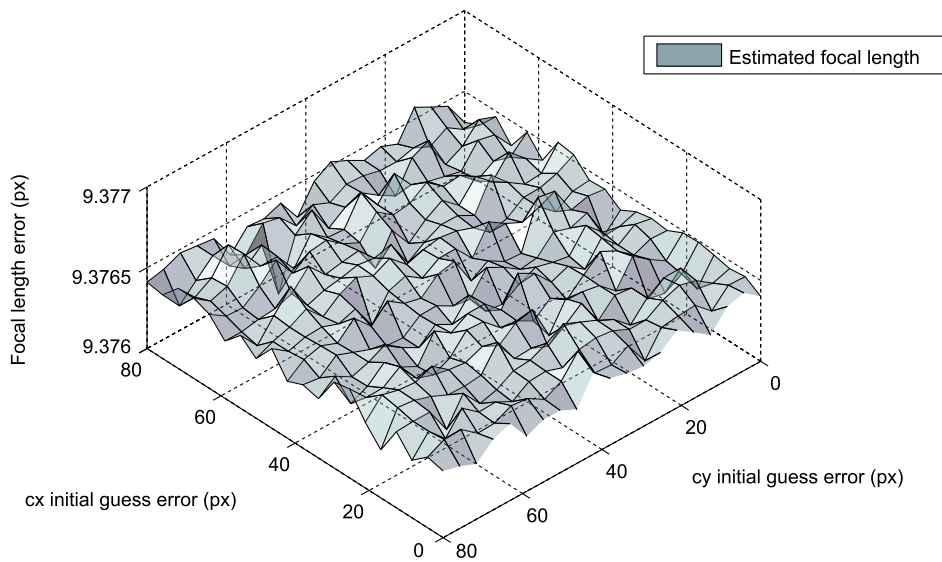


Figure 5.5: Focal length estimation error after the optimization varying the initial optical center guess.

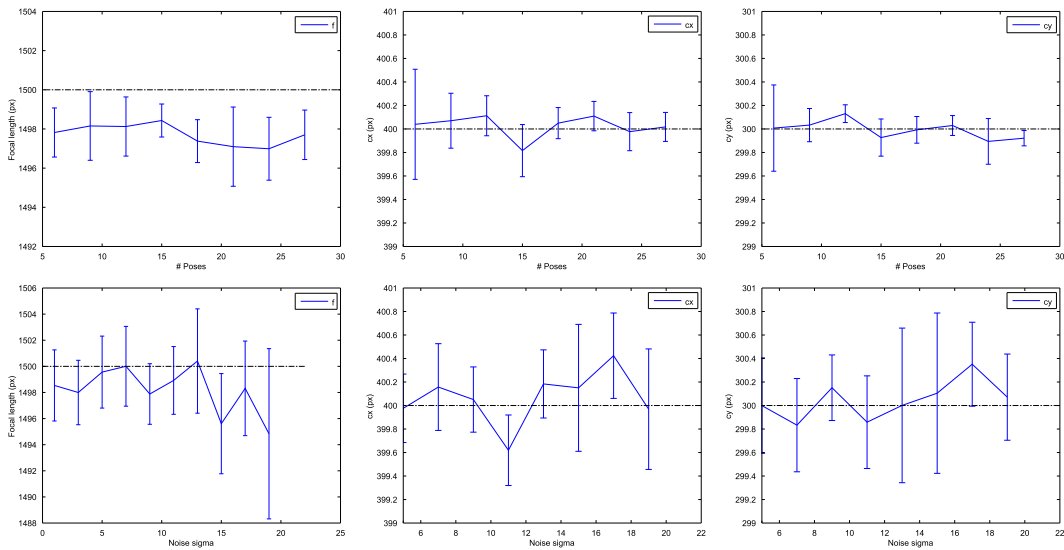


Figure 5.6: Estimated focal length (Left), optical center x (Center) and optical center y (Right) with respect of the number of poses (Top row) and noise (Bottom row). Ground truth is indicated with a dashed line.

5.5 Conclusion

In this chapter we described a novel method to estimate the focal length and the optical center of a pinhole camera by exploiting the projective features of a generic set of coplanar circles. Since circular coplanar features are quite common in human-made environments, we believe that such tool should come in handy when camera calibration must be performed on-the-field and a known calibration target cannot be used. For instance, it may be the case that the camera optical geometry changes during the operation, for example when varying the focal length by using zoom lenses or if the device undertake mechanical stress (i.e vibrations, shocks, etc.). Another scenario that makes our approach appealing rise when we need to calibrate a camera that was not originally meant to be used for computer vision applications, like performing structure-from-motion from publicly available on-line pictures.

To letting the system even more robust, we developed an inlier selection process to discriminate a good cluster of coplanar circles from each image. Moreover, this approach do not require delicate tuning of parameters and produces a fair initial intrinsics calibration to be used as a good starting point for the subsequent refinement.

After the initial guess, a refinement of such estimation is performed by minimizing an energy function that accounts for the eccentricity variation of the ellipses with respect to the pinhole model parameters. Specifically, through a non-linear gradient descent, we modify the initial estimation taking into account the energy gradient that can be analytically computed inside the optimization domain.

II

Model-free camera calibration

6

Can an Unconstrained Imaging Model be Effectively Used for Pinhole Cameras?

Traditional camera models are often the result of a compromise between the ability to account for non-linearities in the image formation model and the need for a feasible number of degrees of freedom in the estimation process. These considerations led to the definition of several ad hoc models that best adapt to different imaging devices, ranging from pinhole cameras with no radial distortion to the more complex catadioptric or polydioptric optics.

In this chapter we propose the use of an unconstrained model even in standard central camera settings dominated by the pinhole model, and introduce a novel calibration approach that can deal effectively with the huge number of free parameters associated with it, resulting in a higher precision calibration than what is possible with the standard pinhole model with correction for radial distortion. This effectively extends the use of general models to settings that traditionally have been ruled by parametric approaches out of practical considerations. The benefit of such an unconstrained model to quasi-pinhole central cameras is supported by an extensive experimental validation.

6.1 Introduction

The literature has deemed the unconstrained model and related calibration procedures a last resort to be adopted only when traditional approaches fail due to either geometrical or methodological issues. For this reason the pinhole model, augmented with a proper distortion correction, dominates the application landscape whenever its use is feasible. In this chapter we explore the opposite direction. Specifically, we ask ourselves (and the reader) whether even the most traditional perspective camera could benefit of the adoption of a non-parametric model. For this to be the case, the calibration must be both effective and reasonably easy to perform.

In the following sections we briefly describe our generic model (which is indeed pretty standard) and we introduce a practical calibration method. The impact on the calibration accuracy with respect to the pinhole model is evaluated with a wide set of experiments. Finally, aspects and implications related to the use of an unconstrained model with common computer vision tasks are discussed.

6.2 Imaging Model and Calibration

In this work we explore the possibility of using a fully unconstrained camera model even for central cameras. In the proposed model each pixel is associated with the light ray direction from the object to where the ray hits the optics, rendering completely irrelevant how the optics bend the light to hit the CCD. This ray can be formalized as a line in the Euclidean space which, in the unconstrained model, is independent on the lines assigned to the other pixels and completely free with respect to direction and position. Under these assumptions, pixels cease to hold a precise geometrical meaning and they become just indexes to the imaging rays, having completely hidden the path that the ray has to go through inside the optics to hit the right cell in the CCD.

In what follows, index i ranges over camera pixels. The ray associated with camera pixel i can be written as $\vec{r}_i = (\vec{d}_i, \vec{p}_i)$, where $\vec{d}_i, \vec{p}_i \in \mathbb{R}^3$ represent direction and position of the ray respectively (see Figure 6.1). These vectors satisfy $\|\vec{d}_i\| = 1$, (normalized direction) and $\vec{d}_i^T \vec{p}_i = 0$ (orthogonal position vector). Any point \vec{x} in the ray \vec{r}_i satisfies the parametric equation $\vec{x} = \vec{d}_i t + \vec{p}_i$ for some $t \in \mathbb{R}$.

This model has 4 degrees of freedom per pixel, resulting in several million parameters to be estimated for current cameras, a dimensionality that is beyond the possibilities of the most commonly used calibration processes. Note, however, that the ray independence assumptions allows for (conditionally) independent estimation of each ray, allowing us to measure the convergence of the estimate as a measure of per-pixel observation. One problem with the commonly used target-based calibration systems (be them chessboard-based, dot-based, or based on any other pattern) is that they provide sparse localization points, resulting in rather low numbers of per-pixel observations. We argue that it is this situation that forces the use of a low dimensional imaging model as the unconstrained model would not converge for every ray, even with a very

large number of poses.

We propose to solve this problem by providing dense localization on the target, thus resulting in one observation per pixel in each pose of the calibration target. This dense calibration target is obtained through the use of structured light patterns on a normal LCD display, allowing us to assign to each camera pixel the 2D coordinate in the target planar reference frame of the location where the ray associated to the pixel hits the target. In particular, we use phase coding with the number-theoretical phase unwrapping approach presented by Lilienblum and Michaelis [118] to encode both the horizontal and vertical coordinate of each pixel of the target display. Each camera pixel integrates this signal over all the incoming rays incident that CCD cell, resulting in a high precision sub-pixel localization of the average incident ray and thus a localization in the target's surface with a precision that is sub-pixel with respect to the target's pixel dimension.

Let index s range over the calibration shots, with $\Theta_s = (\mathbf{R}_s, \vec{t}_s)$ we denote the pose parameters of the calibration target in shot s , transforming the target's coordinate system onto the camera coordinates. The \vec{u}_s , \vec{v}_s , and \vec{n}_s base vectors of the target coordinate system expressed in the camera coordinate system correspond to the first, second and third columns of \mathbf{R}_s respectively, i.e., $\mathbf{R}_s = (\vec{u}_s \vec{v}_s \vec{n}_s)$.

Further, let $\mathbf{Co}_i^s \in \mathbb{R}^2$ denote the code (target 2D location) measured at camera pixel i in shot s , while with $\mathbf{Ce}(\vec{r}_i | \Theta_s) \in \mathbb{R}^2$ we denote the expected code at pixel i , given ray \vec{r}_i and target pose Θ_s (see Figure 6.1). Ignoring possible refraction effects on the monitor's surface this corresponds simply to the surface coordinates (u, v) of the intersection between the ray and the target plane. To add the effects of refraction, we need to add a correction term accounting for Snell's law. For the moment we will concentrate on the refraction-less case, and add the refraction term and analyze its effects on Subsection 6.2.4. In the refraction-less case, we have:

$$\mathbf{Ce}(\vec{r}_i | \Theta_s) = \mathbf{P}_{\vec{u}\vec{v}} \left(\vec{d}_i t_{\text{int}} + \vec{p}_i \right) = (\vec{u}_s \vec{v}_s)^T \left(\frac{\vec{n}_s^T (\vec{t}_s - \vec{p}_i)}{\vec{n}_s^T \vec{d}_i} \vec{d}_i + (\vec{p}_i - \vec{t}_s) \right), \quad (6.1)$$

where $\mathbf{P}_{\vec{u}\vec{v}}$ denotes the projection onto the (u, v) planar coordinates, and

$$t_{\text{int}} = \frac{\vec{n}_s^T (\vec{t}_s - \vec{p}_i)}{\vec{n}_s^T \vec{d}_i} \quad (6.2)$$

is the intersecting parameter for the equation of ray \vec{r}_i , i.e., the value such that $\vec{d}_i t_{\text{int}} + \vec{p}_i$ lies on the target plane.

Under this setup the calibration process can be cast as a joint estimation of the $\vec{\mathbf{r}}$ and Θ parameters.

6.2.1 Least Squares Formulation

We express the calibration process as a generalized least squares problem [102]. Generalized least squares is a technique for estimating the unknown parameters from a

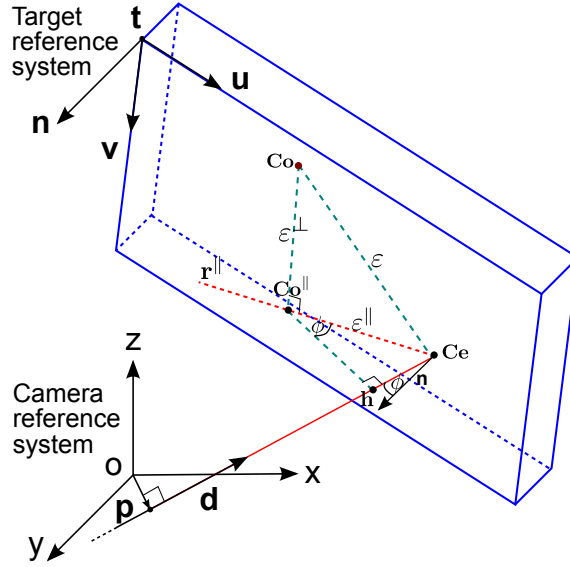


Figure 6.1: Schema of the general camera model and calibration target described in this chapter. Note that the Mahalanobis distance between observed and expected code is equal to the 3D distance of observed code to the ray.

(non-)linear model in presence of heteroscedasticity, i.e., where the variance of the observations are unequal and/or correlated. In such a situation ordinary least squares can be statistically inefficient, while the general least squares estimator provides an efficient estimator. The approach accounts for heteroscedasticity by normalizing the residuals through the inverse of the covariance of the measurements Σ , thus minimizing the sum of the squared Mahalanobis lengths of the residuals:

$$\sum_k (y_k - f(x_k, \vec{\theta})) \Sigma_k^{-1} (y_k - f(x_k, \vec{\theta}))^T. \quad (6.3)$$

Let $\epsilon_i^s = \mathbf{Co}_i^s - \mathbf{Ce}(\vec{r}_i | \Theta_s)$ be the code residuals, then the generalized least squared estimate of the rays and pose parameters \vec{r}, Θ is:

$$(\hat{\vec{r}}, \hat{\Theta}) = \operatorname{argmin}_{\vec{r}, \Theta} \sum_{i,j,s} (\epsilon_i^s)^T (\Sigma_i^s)^{-1} \epsilon_i^s, \quad (6.4)$$

where Σ_i^s is the (conditional) error covariance matrix under the given pixel-pose combination.

In this context the main source of heteroscedasticity derives from the directional correlation of code errors when rays hit the target plane at an angle. In fact, let ϕ be the angle between the ray direction \vec{d} and the normal to the target \vec{n} , and let $\vec{r} \parallel_i^s = \frac{(\vec{u}_s \vec{v}_s)^T \vec{d}_i}{\|(\vec{u}_s \vec{v}_s)^T \vec{d}_i\|}$ be the direction of the ray projected onto the target's planar coordinates, then measurement errors will be amplified by a factor of $\frac{1}{\cos^2 \phi}$ along $\vec{r} \parallel_i^s$ due to the effects of the grazing angle, while will remain unchanged along the orthogonal direction.

Hence, we obtain

$$\begin{aligned} (\Sigma_i^s)^{-1} &= I + (\cos^2 \phi_i^s - 1) \vec{r} \|\vec{r}\|_i^s \|\vec{r}\|_i^s{}^T \\ &= I - (\vec{u}_s \vec{v}_s)^T \vec{d}_i \vec{d}_i^T (\vec{u}_s \vec{v}_s). \end{aligned} \quad (6.5)$$

In the standard pinhole model the effect of eliminating the correlation of the errors is obtained by re-projecting the residuals onto the image plane, i.e., by minimizing the reprojection error rather than the planar displacement over the target. In this sense, normalizing over the inverse error covariance is as close as one can get to the minimization of the reprojection (or geometrical) error in an unconstrained system that loses any direct geometrical connection between rays and pixels.

It is worth mentioning that the reprojection error in the pinhole model accounts for another source of heteroscedasticity, i.e., a change of scale in the covariance as the targets moves away from the camera as a consequence of foreshortening. In the unconstrained model this would imply an unknown scale term on the covariance Σ that would require inter-ray interaction in order to be estimated, complicating the least square formulation and hindering our ability to estimate the parameters effectively and/or with reasonable computational effort. For this reason in the formulation we are ignoring the depth-related change of scale in the error variance. We note, however, that this effect is relatively limited, since it would induce a variation in the scale of the error covariance proportional to $\frac{\Delta z}{z}$ (depth variability over average depth) which in our setup reduces to a variability within approximately $\pm 10\%$ of the average.

To optimize the least squares formulation efficiently, we make use of the conditional independence of the ray parameters \vec{r} given the poses Θ and of the poses given the rays. We do this by performing a two-step optimization process in which we alternatively optimize all the ray parameters in parallel keeping the pose parameters fixed, and then optimize the poses keeping the rays fixed. This way, the large scale estimation part, i.e., the optimization of the ray parameters, becomes massively parallel and can be computed very efficiently in GPU. In our experiments the optimization process is initialized with the normal pinhole model with polynomial radial distortion.

6.2.2 Ray Calibration

As we fix the pose parameters, all the rays depend only on the observed coordinates \mathbf{Co} associated with each of them and can be estimated independently. Further, with the pose parameters at hand, these observed 2D coordinates can be transformed into 3D points in the camera coordinate frame. As can be seen in Figure 6.1, given a ray \vec{r} intersecting the target plane at 2D coordinate \mathbf{Ce} , we can divide the residual $\varepsilon = \mathbf{Ce} - \mathbf{Co}$ into the orthogonal vectors $\varepsilon^\parallel = \mathbf{Ce} - \mathbf{Co}^\parallel$ and $\varepsilon^\perp = \mathbf{Co}^\parallel - \mathbf{Co}$, where ε^\parallel is parallel to \vec{r}^\parallel . Clearly, since ε^\perp is orthogonal to the plane spanned by \vec{d} and \vec{n} , the point in \vec{r} closest to \mathbf{Co} is also the one closest to \mathbf{Co}^\parallel . Further, let \vec{h} be this point, we have

$$\|\vec{h} - \mathbf{Co}\|^2 = \|\vec{h} - \mathbf{Co}^\parallel\|^2 + \|\varepsilon^\perp\|^2. \quad (6.6)$$

It is easy to show that, $\|\vec{h} - \mathbf{Co}\| = \cos\phi\|\varepsilon\|$, where ϕ is the angle between \vec{d} and \vec{n} . Hence, the squared distance between \vec{r} and \mathbf{Co} equals

$$d^2(\vec{r}, \mathbf{Co}) = \cos^2\phi\|\varepsilon\|^2 + \|\varepsilon^\perp\|^2 = \varepsilon^T \Sigma^{-1} \varepsilon, \quad (6.7)$$

thus the generalized least squares formulation with respect to the target coordinates corresponds to the standard linear least squares with respect to the 3D points associated with each ray. The linear least squares problem is then solved by a ray with parametric equation $\vec{x} + \vec{w}t$, where $\vec{x} = \frac{1}{n} \sum_{i=1}^n \vec{x}_i$ is the barycenter of the observed 3D points, and \vec{w} is the eigenvector of their covariance matrix corresponding to the smallest eigenvalue.

6.2.3 Estimation of the Poses

The second step in the alternating calibration process is the optimization of the target poses keeping the rays fixed. Also in this step we can make use of a conditional independence; in fact, with the rays fixed, the pose parameters from different shots become independent and can be optimized in parallel. Further, just like in the ray calibration step, we make use of the equivalence between the Mahalanobis distance over the 2D core errors $\mathbf{Ce} - \mathbf{Co}$ with the 3D Euclidean distance between observed position and ray. In this situation the estimation of pose Θ_s reduces to the search for the rigid transformation that minimizes the distance between the (transformed) observations $\Theta_s \mathbf{Co}_i^s$ and the ray \vec{r}_i . This minimization problem shares several similarities with surface alignment, where we seek to find the rigid transformation that minimizes the Euclidean distance between a set of points on a surface to points on another surface.

It is not surprising, then, that a straightforward modification of the Iterative Closest Point (ICP) algorithm [33] can be used to find a (local) optimum for our problem. The modification lies in how the closest points are sought: In our context instead of searching for the closest point in a single given 2D surface common for all the points, we select the closest point in the unique 1D ray associated with the given observation. The iterative selection of all the closest points for all the pixels in each shot and application of Horn's alignment algorithm [91] converges to a locally optimal pose.

Clearly, given the generality of the imaging model, the optimization problem is not convex, thus we cannot guarantee a global optimum like in the case of the pinhole model. However, under the assumption of a central quasi-pinhole camera, we can obtain a very good initialization and be confident about the quality of the final solution found.

6.2.4 Accounting for Refraction

In our formulation we have ignored the possible effects of refraction on the monitor's surface.

In [161] it was shown that refraction had a small but noticeable effect when calibrating using LCD displays, so we extended the least square formulation to incorporate

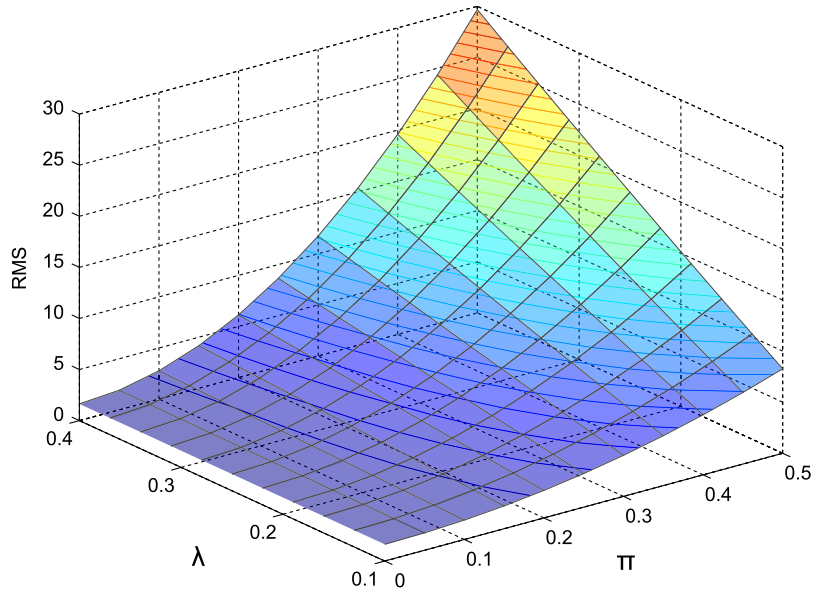


Figure 6.2: Effect of refraction correction for different values of the refraction parameters.

Snell's law of optical refraction in order to assess and correct its effects. To this end, we added two global parameters λ and μ representing respectively the (inverse) refractive index between air and the transparent layer in front of the LCD, and the depth of the layer. According to Snell's law, a ray \vec{r} hitting the surface at an angle ϕ with the normal \vec{n} , will be refracted inside the transparent layer at an angle ψ satisfying $\sin \psi = \lambda \sin \phi$, hitting the reflective layer at target coordinates $\mathbf{C}\mathbf{e} + \Delta\mathbf{C}\mathbf{e}$ with

$$\Delta\mathbf{C}\mathbf{e} = \vec{r}^{\parallel} \mu \tan \psi = \mu \frac{\lambda \sin \phi}{\sqrt{1 - \lambda^2 \sin^2 \phi}}. \quad (6.8)$$

The addition of refraction adds a non-linearity in both the ray and pose estimation that breaks the conditional independence assumption at the basis of our approach. We solve this by adopting a fixed point approach, reiterating the least squares estimations (both in the ray and pose estimation phases) using the refraction shift $\Delta\mathbf{C}\mathbf{e}$ computed based on the previous rays and poses.

Figure 6.2 shows the effect of the refraction parameters on the final root mean squared error (RMS) of the calibrated camera. From the plot we can see clearly that the minimum is attained in the refraction-less case ($\mu = 0$ or $\lambda = 0$), thus pointing to a negligible effect of refraction for the unconstrained model as opposed to what was reported in [161] for the pinhole model. It must be said that our experiments with the pinhole model gave inconsistent results, exhibiting error reduction as reported by

Schmalz et al. when using few points to perform the calibration, and showing no effect when using more points. This can be explained by the fact that target shots are mostly frontal to the camera and thus the effect of refraction is mostly radial. Hence, adding a refraction term changes (possibly enlarging) the space of radial functions used for eliminating distortion with the pinhole model. Using more points to perform the calibration constrains the model more, resulting in no additional advantage. Our model, not having an explicit radial distortion term, is not affected by this phenomenon.

To push this finding to the extreme, we placed a 0.75mm glass layer (the front glass of a photo frame) in front of the LCD display in an attempt to produce a much stronger refraction effect. Even in this condition no effect could be measured both in the unconstrained and in the dense pinhole case.

While the estimation of the λ and μ parameters could be incorporated into the calibration process as a third stage performing gradient descent over the parameter λ , the fact that we could not observe any effect moved us to ignore refraction altogether in the experimental evaluation of our approach.

6.3 Working with the Unconstrained Camera

By alternating the two estimation process we obtain the generalized least squares estimation of both rays and poses, and with that a full calibration of the unconstrained camera model. However, for the unconstrained model to become an effective alternative to the pinhole model several problems must be solved. In particular, if we want to use the model for high precision 3D reconstruction, we need at the very least an effective algorithm for stereo calibration as well as a way to interpolate rays at non-integer coordinates. Potentially we also need a wider set of geometrical and algorithmic tools that are either straightforward or well studied for the pinhole model. In fact, the parametric nature of the pinhole model offers a direct solution the interpolation problem, while there is a ample body of work on how to estimate the motion between two calibrated cameras. As a matter of fact, the pinhole model also allows for useful processes like rectification that even conceptually does not have a counterpart in the unconstrained model. However, arguably any measurement or reconstruction process can be reformulated based on only extrinsic calibration and ray interpolation, which incidentally is the minimal requirement to perform triangulation effectively.

For the stereo calibration we adopt a quite crude approach: We take several shots of our active target with both cameras and use the modified ICP algorithm to estimate the poses for each camera. As usual we initialize the poses with the pinhole model to guarantee convergence. With the poses of the first camera at hand, we can construct a set of 3D points \vec{x}_i^s in the first camera's coordinate system where ray \vec{r}_i intersects the target plane at shot s . Further, given the codes Co_i^s observed in pixel i at shot s and the pose Θ'_s of the target according to the second camera, we can compute the points \vec{y}_i^s which represent the points \vec{x}_i^s in the coordinate system of the second camera. once we have computed these pairs of points for all camera pixel and shots, we can use Horn's

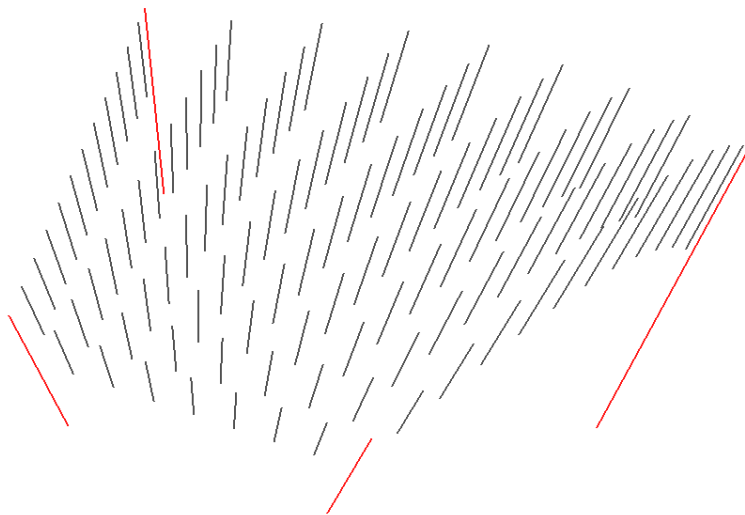


Figure 6.3: Manifold interpolation of free rays.

alignment algorithm [91] to estimate the transformation between the two coordinate systems. Clearly this approach has limits, as it does not attempt to reduce the sources of heteroscedasticity, but merely averages over the pose estimations that are by their nature affected by noise, while a more principled approach should pose the problem as global optimization over the observables. However, it is enough to test the feasibility of the unconstrained model and, in conjunction with the unconstrained model, does provide, as will be shown in the experimental section, more precise and repeatable 3D measures than what can be obtained with state-of-the-art approaches based on the pinhole model.

6.4 Rays interpolation

One reason that lets the pinhole camera model so effective in practice is that exists a simple mapping between any point p in the normalized image plane and the corresponding ray entering the camera. Indeed, since all rays are forced to pass through the optical center o , the unique straight line connecting o and p defines the light ray that the camera can sample at p . Consequently, many fundamental problems like 3D point triangulation can be solved in a multitude of simple and powerful ways [86].

Diversely, if we model our camera as a generic sparse bundle of probing rays, there is no trivial way to recover the ray entering our imaging device at any (possibly sub-pixel) CCD point p . To be honest, there is neither a concept of CCD or image plane but just some existing calibrated rays in space each one sampling an intensity value or,

if we use a structured-light system, a two-dimensional phase code. Still, there is the basic need to estimate a ray \vec{r} at any point in space assuming a concept of continuity in the camera rays bundle hence expressing \vec{r} via some sort of interpolation of a set of known calibrated rays.

Despite the specific goal to achieve while estimating an unknown ray \vec{r}_ℓ , we need to tackle two equally important problems. First, we need to select a non-empty set of already known rays together with a vector of weights \vec{w} that affects how strongly a specific ray is involved in the estimation of \vec{r}_ℓ . Second, we need to define an interpolation function that performs the estimation possibly exhibiting some physically meaningful properties. We start by proposing a solution for the latter problem in Sec.6.4.1 while the former is discussed in Sec.6.4.2.

6.4.1 Rays manifold interpolation function

Let $R_d = \{\vec{r}_i\}$ a set of n known camera rays, and $\vec{w} = (w_1, \dots, w_n) \in \mathbb{R}^n, \sum_{i=1}^n w_i = 1$ a convex combination of weights.

We start by reasoning on the simple case in which $n = 2$ and $\vec{w} = (1 - t, t), t \in [0 \dots 1]$. Any *valid* interpolation function $\vec{r}_\ell = \Psi(\vec{r}_1, \vec{r}_2, t)$ should be such that: for $t = 0$ and $t = 1$ it returns the original \vec{r}_1 and \vec{r}_2 respectively (6.9), the interpolation is independent on the order of the transformations (6.10) and, finally, a valid ray is obtained for each possible value of t .

$$\Psi(\vec{r}_1, \vec{r}_2, 0) = \vec{r}_1, \Psi(\vec{r}_1, \vec{r}_2, 1) = \vec{r}_2 \quad (6.9)$$

$$\Psi(\vec{r}_1, \vec{r}_2, t) = \Psi(\vec{r}_2, \vec{r}_1, 1 - t) \quad (6.10)$$

We pose the ray interpolation problem in terms of rigid motions blending. Let $K \in SE(3), K(\vec{r}_a) = \vec{r}_b$ the rigid motion that transforms a ray \vec{r}_a into \vec{r}_b . Such motion can be decomposed into 3 distinct motions:

1. A rotation Qr_K (around an arbitrary axis) that aligns \vec{d}_a and \vec{d}_b . i.e. $\vec{d}'_a = R_K(\vec{d}_a) = \vec{d}_b$
2. A translation Qt_K that lets the ray \vec{r}_a coincide with \vec{r}_b . i.e. $\vec{p}'_a = T_K(\vec{p}_a) = \kappa \vec{d}_b + \vec{p}_b, \kappa \in \mathbb{R}$.
3. Any rotation around $\vec{d}'_a = \vec{d}_b$ and a translation along the same direction.

Note that the first two operations are the ones required to effectively transform the ray while the latter is invariant to the transformation since it rotates and the translate the resulting ray along its direction (i.e. the set of 3D points composing the ray is not changing).

At this point, the function Ψ can be implemented in terms of any rigid-motion blending function Φ that interpolates K according to t . Still, we are left with the problems of choosing a specific transformation K (for instance, there are many alternative

axis for Qr_K) and motion interpolation Φ . Among the infinite set of possible choices, it seems reasonable trying to minimize the path traced by the points of \vec{r}_a while moving toward \vec{r}_b according to Ψ .

For the blending function Φ , we choose the *Dual-quaternion Iterative Blending* (DIB) that interpolates a roto-translation in terms of a screw motion [104], exhibiting many useful properties such being *constant speed*, *shortest path* and *coordinate system independent*. Therefore, we describe all the roto-translations as unitary dual quaternions and measure the path length of points undertaking a rigid motion with the length of the screw motion involved. Concerning the choice of K we observe that, when applied to rays, pure translations move all the ray points through a path as long as the translation vector's length whereas, for pure rotations, the farther a point is from the rotation axis the longer its path would be. Consequently, it makes sense to give more weight to motions with smaller rotation angles with respect to the amount of translation. In other terms, given two motions K_1 and K_2 , we establish a total ordering so that $K_1 < K_2$ if and only if K_1 has smaller rotation angle or, if the two angles are the same, the translation vector of K_1 is shorter than the one of K_2 .

From the aforementioned statement is straightforward to see that the best possible rotation angle is the one between the two vectors \vec{d}_a and \vec{d}_b (i.e. $\text{acos}(\vec{d}_a^T \vec{d}_b)$) that rotates the first ray around the axis given by $\vec{d}_a \times \vec{d}_b$. When the rotation angle and axis is chosen, the minimum translation Qt_K is the one moving \vec{r}_a according to a vector T orthogonal to $\vec{d}_a = \vec{d}_b$ whose length is equal of the distance between the two rays. In other terms, the best translation is the vector that connects the two nearest points \vec{s}_1 and \vec{s}_2 lying on \vec{r}_1 and \vec{r}_2 respectively. To summarize, given two rays, we choose K as:

1. The rotation R_K around the axis $\vec{d}_a \times \vec{d}_b$ with angle $\text{acos}(\vec{d}_a^T \vec{d}_b)$
2. The translation $T_K = \vec{s}_2 - \vec{s}_1$

as the minimum motion transforming the first into the second.

Since we only defined the interpolation in terms of pair-wise motions, we generalize the case of $n > 2$ through a simple iterative procedure. Specifically, we start with an initial estimate of \vec{r}_ℓ as the linear combination $\vec{r}_\ell = \sum_{i=1}^n w_i \vec{r}_i$ followed by a re-projection on the rays manifold (i.e. a normalization of \vec{d}_i and a reparametrization of \vec{p}_i such that $\vec{d}_i^T \vec{p}_i = 0$). Then, we compute the rigid transformations $K_{\ell,i}$ as the screw motion between \vec{r}_ℓ and each \vec{r}_i according to the procedure stated before. Once computed, all the $K_{\ell,i}$ are averaged via DIB with the weights \vec{w} to obtain K_{avg} . Finally, K_{avg} is applied to \vec{r}_ℓ to obtain a better estimate, and the procedure is repeated until the length of the path induced by K_{avg} is less than a fixed threshold.

6.4.2 Selecting the interpolation data

Once an interpolation strategy is defined, we are left with the problem to select a meaningful set of rays (with associated weights) to perform the interpolation. In this case,

since we don't pose assumptions on the neighborhood relations among the rays, we can only rely on the code observed during the calibration. Specifically, we select the k -nearest (in terms of code distance) rays to the sought code c_g with their respective code distances $d_1 \dots d_n$. Then, we perform the interpolation by using the inverse of the squared distance as interpolation weights¹:

$$w_i = \frac{\frac{1}{d_i^2}}{\sum_{i=1}^k \frac{1}{d_i^2}}, \quad i = 1 \dots k$$

6.5 Experimental Evaluation

In order to evaluate the effectiveness of our calibration approach and the usefulness of the unconstrained model for high-precision vision applications with quasi-pinhole central cameras, we performed several calibration with our method based on a training set of 40 shots of the active target, and tested them on a test set composed of 40 shots.

For the active target we used an LCD display with a resolution of 1280x1024 pixels, while we used a professional entry level 1 megapixel computer vision camera with variable focal length optics set close to the shortest available length in order to have noticeable, but not extreme distortion. The rays and poses were initialized performing a pinhole calibration using the OpenCV library [36] and adopting a 5th order polynomial model for the radial distortion.

Figure 6.4 plots the root mean squared error between expected and observed codes as a function of the number of iterations of the calibration algorithm. While the iteration of the calibration procedure was performed on the training set, the computation of the error was performed on the test set by running only the pose estimation without changing the rays. The top figure plots the error averaged over all the pixels and poses, and clearly shows that the estimated model exhibits an order of magnitude lower error than the pinhole model, which is the initialization model and thus the first entry in the plot. The images in the bottom row display the error for each pixel, averaged over the poses. The leftmost image refers to the initial pinhole model, while the middle and right image refer to the model after 2 and 21 iterations of the calibration procedure. Red pixels represent high error, while blue pixels represent low error. It is immediately apparent that there is still some residual structured error in the pinhole model that the polynomial radial distortion term was not able to correct, resulting in radial waves of high and low error corresponding to areas where the estimated polynomial fits more or less accurately the model. Apart from the radial distortion, however, there is still some error that is not radially symmetric, as can be seen by the different levels of blue and green around the low-error ring. The unconstrained model, on the other hand,

¹We safely assume that all the distances are greater than zero. Otherwise, the interpolation itself is meaningless since we already know the exact ray for a specific code

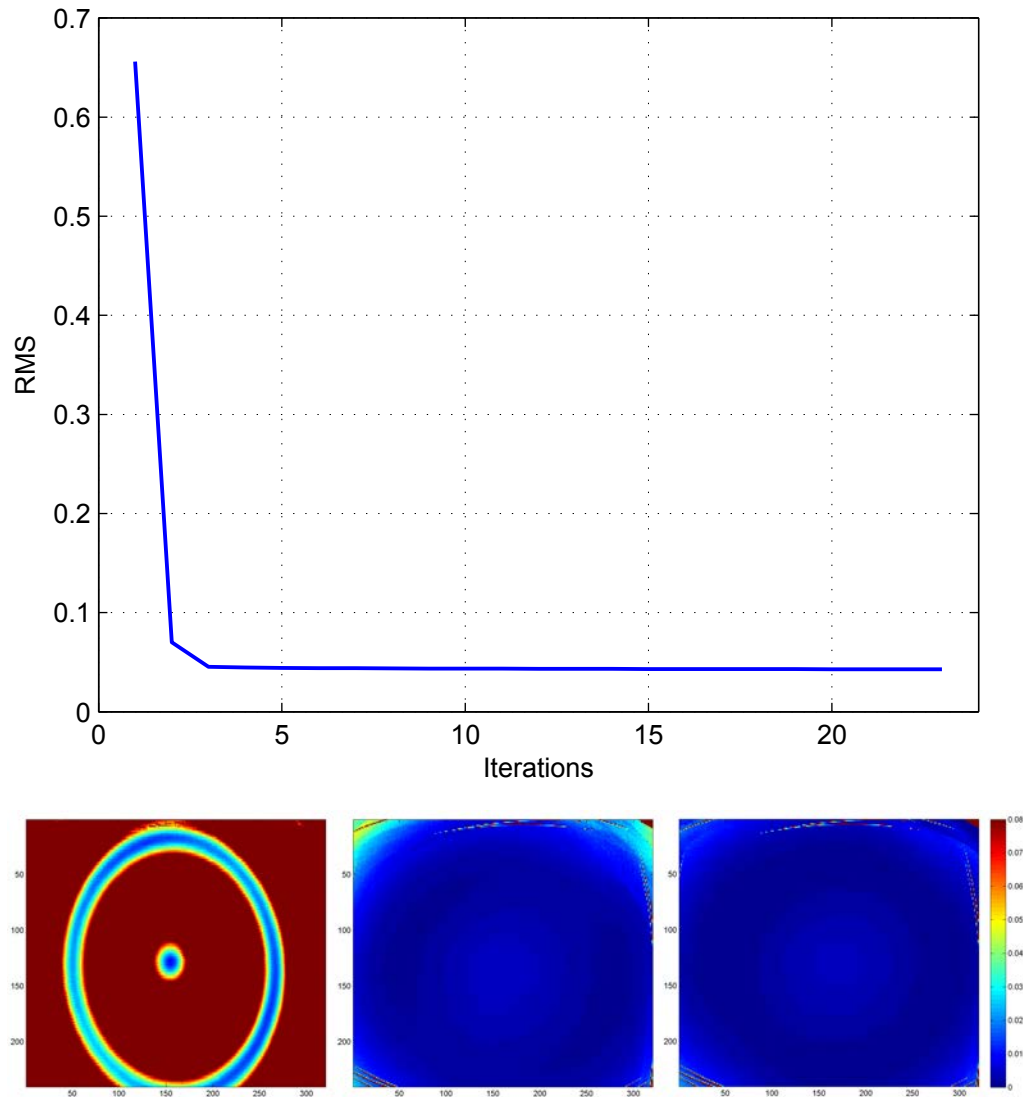


Figure 6.4: Root mean squared error between expected and observed codes as a function of the number of iterations of the calibration algorithm. The top plot shows the error averaged over all the pixels and poses, while the bottom pictures show for each pixel the error averaged over all the poses at iteration 0 (pinhole model), 2, and 21.

986. Can an Unconstrained Imaging Model be Effectively Used for Pinhole Cameras?

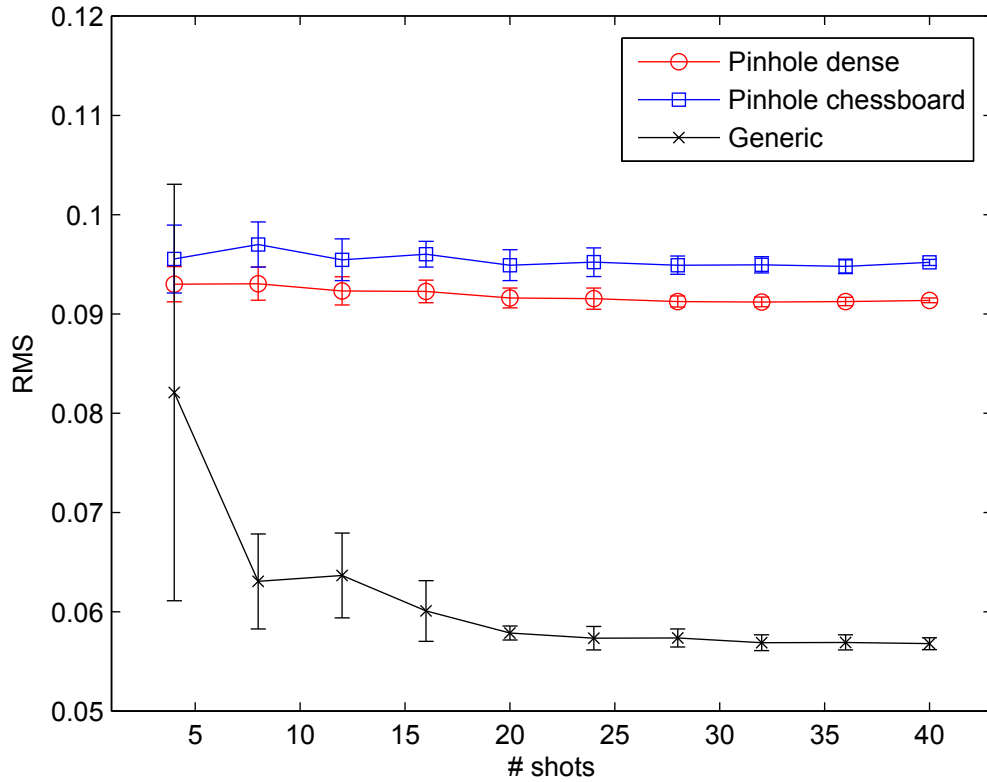


Figure 6.5: Comparison of the error obtained with the pinhole model calibrated with a chessboard and dense target, and with our calibration approach for the unconstrained model.

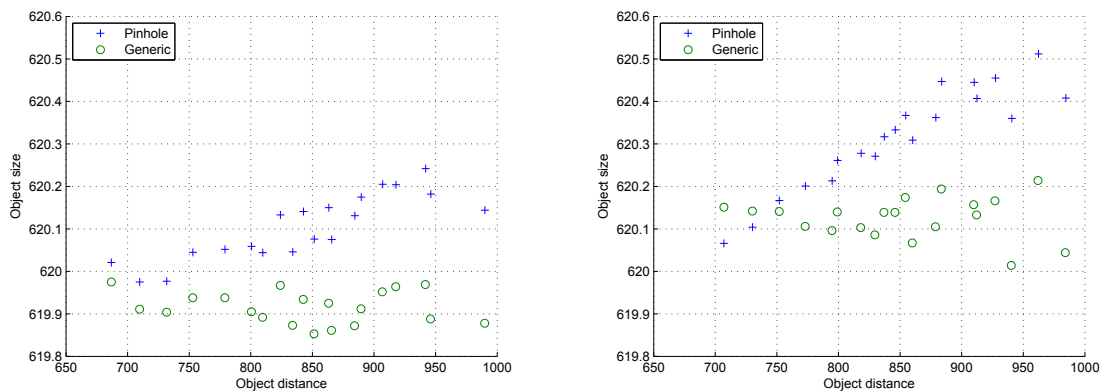


Figure 6.6: Scatter plot of the measured distances between pairs of points taken in the top part (left) and bottom part (right) of the target. The points where 620 target pixels apart.

not only dramatically reduces the error, but also mitigates the spatial coherency of the error.

Figure 6.5 plots the RMS error on the test set obtained with our model as we increase the number of poses in the training set. The plot is compared against the results obtained with a pinhole model calibrated with a standard chessboard pattern and using the same LCD-based active dense target used to calibrate the unconstrained model. We can see that the dense target offers a marginal advantage over the chessboard target for the pinhole model. Note, however, that this is most likely due to the increased precision in the localization of the point offered by the phase shift encoding rather than to the increase in number of points. In fact, no advantage can be seen in adding more shots, since the low dimensional model is already well constrained with a very limited number of poses. The unconstrained model, on the other hand, clearly needs more shots to fully estimate the huge number of free parameters, exhibiting a very large variance when calibrated with few shots, and only really settling down to a precise estimation when at least 20 target points are observed for each pixel. However, while exhibiting large variance, the unconstrained model has a lower average RMS error even with as few as 4 shots, reaching an error when estimated with a sufficient number of shots that is approximately an order of magnitude lower than what can be obtained with the pinhole model.

In order to assess the advantage that the unconstrained model offers over the pinhole one in high precision tasks, we performed a very basic 3D measurement task on a calibrated camera pair. Both cameras were calibrated with the pinhole model (both intrinsic and extrinsic parameters) using the OpenCV library [36], and using the proposed approaches for unconstrained camera and stereo calibration. With the calibration parameters at hand, we triangulated two known points on the calibration target in 20 different shots in the test set, and computed their distance as a function of their distance from the camera. Figure 6.6 shows two scatter plots obtained from two different pair of points located in different parts of the target. The target points were, in both cases 620 target pixels apart and the spatial unit of the plot is target pixel width.

From the plot we immediately see that with the pinhole model there is a correlation between depth and measured size, clear indication of an imperfect image formation model, and resulting in a relatively large overall variance of the measure. The unconstrained model, on the other hand, does not exhibit this positional dependency resulting on a much smaller variance in the measurement.

It is worth noting that there is a small bias in the estimated distance. In fact we underestimate the measure of approximately 0.1 pixel out of 620 (about 0.015% or 0.03mm out of a length of approximately 180mm) on the leftmost plot which was drawn from points on the top part of the target, while we overestimate by approximately the same amount on the rightmost plot, taken from points on the bottom part of the target. This can be the result of a non-perfectly rectangular monitor (an error of 0.015% is well within the construction tolerances) and can be seen in the pinhole model as well.

6.6 Discussion

In this part of the thesis we investigated the use of an unconstrained camera model to calibrate central quasi-pinhole cameras for high precision measurement and reconstruction tasks, and provided an effective approach to perform the calibration. The basic ingredient for the calibration process is the use of a dense target which allows us to attain a favorable parameters to measurements ratio, guaranteeing a stable calibration with a limited number of shots, and thus rendering the process operationally not much more complex than standard pinhole calibration with sparse passive targets.

The resulting model can successfully eliminate the spatial coherence of the error, resulting in more precise and repeatable measures than what is achieved with the pinhole model. In fact there is a clear indication that the estimated models are substantially different from a radially undistorted pinhole model. To see this difference, we can visualize how far the rays are from passing through a single point. To this effect, for each pixel and its 4-neighborhood, we define a local pinhole as the point that has minimal sum of squared distances from the five rays.

Figure 6.7 shows the spatial distribution of these local pinholes, sub-sampled for display purposes. It is clear that the distribution is quite different from a random spreading of coincident points, exhibiting a very distinctive spatial coherency, as the points lay quite tightly on a clear manifold linked with the evolute of the calibrated rays.

Clearly, this is only a preliminary analysis, and much work still needs to be done before the unconstrained model can substitute effectively the pinhole model. In particular, more principled approaches to stereo calibration are needed, as well as alternatives for those algorithms that rely on geometrical processes that do not have a counterpart on the unconstrained model. However, we feel that there is much to be gained by moving towards non-parametric camera models, especially in high precision 3D vision tasks.

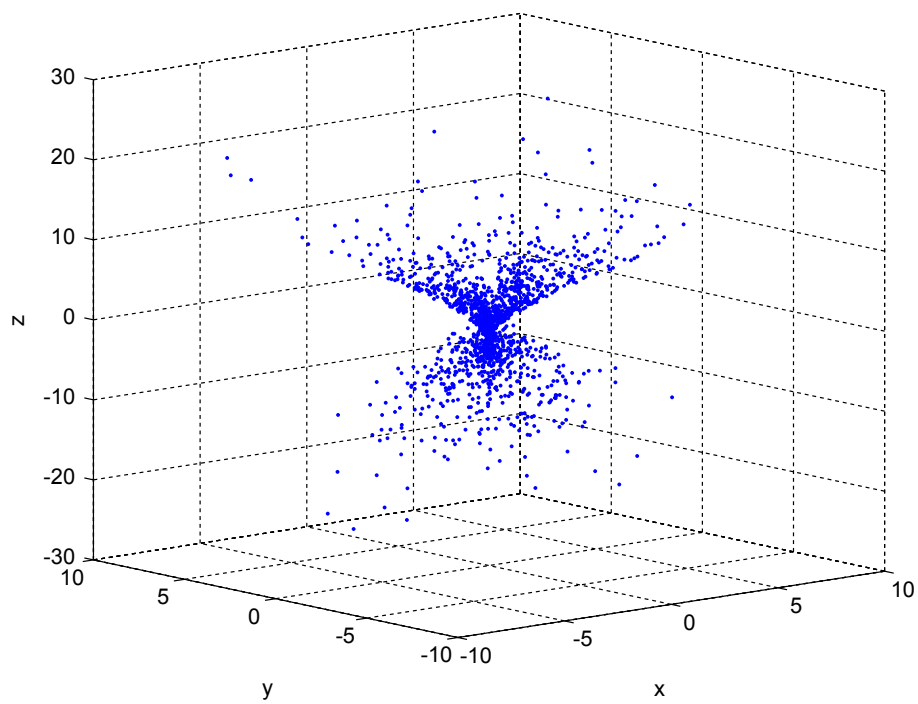


Figure 6.7: Spatial distribution of the local pinholes, i.e., of the closet point to the rays in a local neighborhood. In a pinhole model all the points would coincide. For display purposes the points are sub-sampled.

1026. Can an Unconstrained Imaging Model be Effectively Used for Pinhole Cameras?

7

High-Coverage Scanning through Online Projector Calibration

Many 3D scanning techniques rely on two or more well calibrated imaging cameras and a structured light source. Within these setups the light source does not need any calibration. In fact the shape of the target surface can be inferred by the cameras geometry alone, while the structured light is only exploited to establish stereo correspondences. Unfortunately, this approach requires each reconstructed point to exhibit an unobstructed line of sight from three independent points of views. This requirement limits the amount of scene points that can be effectively captured with each shot. To overcome this restriction, several systems that combine a single camera with a calibrated projector have been proposed. However, this type of calibration is more complex to be performed and its accuracy is hindered by both the indirect measures involved and the lower precision of projector optics.

In this chapter we propose an online calibration method for structured light sources that computes the projector parameters concurrently with regular scanning shots. This results in an easier and seamless process that can be applied directly to most current scanning systems without modification. Moreover, we attain high accuracy by adopting an unconstrained imaging model that is able to handle well even less accurate optics. The improved surface coverage and the quality of the measurements are thoroughly assessed in the experimental section.

7.1 Introduction

Structured light 3D scanners have recently become very popular due to their constantly improving accuracy and affordability. Many of such devices are based on a pair of cameras for point triangulation and on a pattern projection to fix correspondences between observed points. This kind of configuration requires a rather strict condition to hold for a point to be triangulated. Namely, it must be illuminated by the projector and seen by both cameras. The consequences of this requirement are illustrated in Fig. 7.1. In the shown scenario, the relative position of the projector and the camera pair allows to reconstruct only the portion of the surface represented with the thick blue line. In fact the remaining part, depicted with the thin red line, cannot be measured since self-occlusion happens for at least one of the three required lines of sight. By converse, if the system was able to work with just one camera the captured surface would have been much more extended. With this latter scenario, which implies projector calibration, all the non-dashed surface in Fig. 7.1 could have been recovered.

This higher coverage of the imaged surface, combined with the implicit lower costs, asserts the potential usefulness of a camera-projector system. Of course those benefits can be exploited only through an accurate projector calibration which, aptly, is a very well covered topic within literature. Many approaches involve the projection of some special pattern over a known planar target. The underlying idea is to allow the camera to localize the target, while using the pattern to map individual projector pixels.

This is the case, for instance, with the approach recently proposed by Chien et al. [49], where a virtual checkerboard is projected onto a real one and an iterative correction loop is performed on the projector parameters until the camera detect a perfect overlap between the virtual and physical target. Chen and Xi [51] adopt a target made up of a grid of regular dots, which are easy to detect and located accurately. The projector rays are located by using a standard Gray-coded pattern sequence to assign a projector coordinate to each image pixel. Very similar methods are also proposed (among many others) by Huang et al. [95] (using dots and Gray coding), Moreno and Taubin [137] (using checkerboard and Gray coding), and Audet and Okutomi [25] (using physical augmented reality tags together with projected tags).

While the combination of physical and projected target is extensively adopted to calibrate projector-camera systems, several alternative approaches have been explored. A specially crafted light sensor, made with optical fibers, is used by Lee et al. [113]. The individual sensors embedded in the target are used to discover its location by locally decoding a Gray-coded binary pattern projected over them. This way there is no need for the camera to estimate the pose of the target plane. Kimura et al. [106] drops the need for a special reference object, although this advantage is obtained by assuming that the camera has been previously calibrated. The process requires to project onto a plane of unknown pose and uses projective geometry constraints between camera and projector to optimally calibrate the latter. A planar featureless target is also used in the more recent method proposed by Yousef et al. [201]. Here two calibrated cameras are required and passive stereo is performed onto a projected checkerboard in order to es-

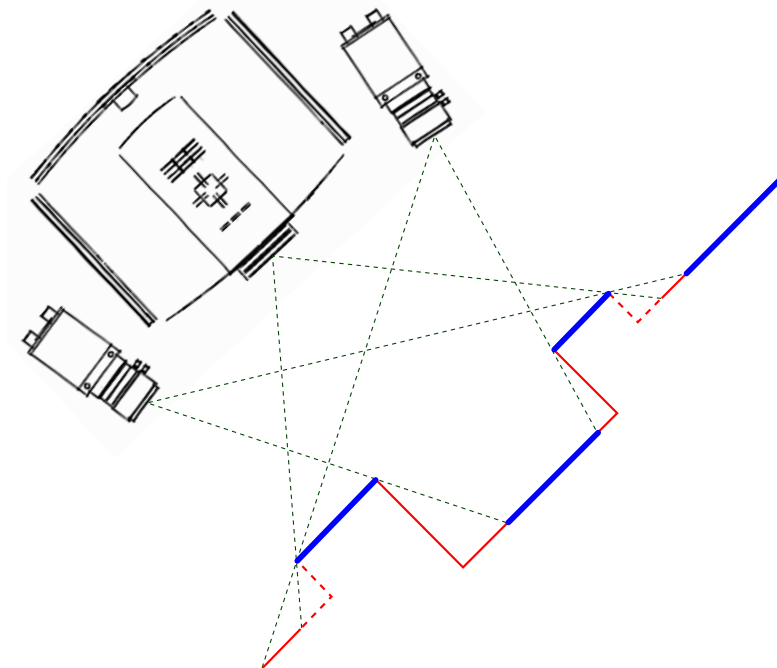


Figure 7.1: The incomplete coverage problem that affects many structured light systems. See the text for details. (image best viewed in color)

establish correspondences. Differently, Okatani and Deguchi [145] demonstrate how to obtain the homographies that relate a plane, a projector and a camera when just the intrinsic parameters are known.

Finally, some methods provide for auto-calibration. That is, the projector calibration happens during the scanning process itself. Most of these approaches are adapted from classical uncalibrated stereo techniques [83, 65, 73] where the second camera is replaced by the projector. A drawback of these methods is that the actual scale of the scene cannot be inferred, however Furukawa and Kawasaki [76] suggested how to augment the system with a laser pointer to recover the correct Euclidean reconstruction. Yamazaki et al. [197] propose an online method that exploits the dense correspondences produced by structured light to compute radial fundamental matrix that is then decomposed into intrinsic and extrinsic parameters for both camera and projector. This way they can account for some lens distortion, however they do not address the scale ambiguity.

With this chapter we introduce a projector calibration technique that does not require to adopt special devices or targets. Neither an explicit calibration procedure is needed. In fact, our approach is an online method that can be performed directly during the normal system usage. Differently from other online methods it is able to automatically recover the scene scale and to deal even with severely distorted lens. However this comes at the price of requiring two calibrated cameras to be paired with the pro-

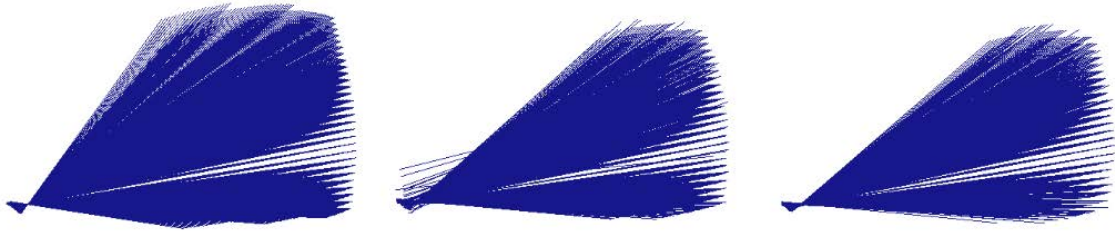


Figure 7.2: The bundles of rays that can be obtained after calibration of the projector using the reconstructed 3D points. In the first image we adopted the pinhole+distortion model. The second and third image show the results obtained using the unconstrained model respectively with and without outlier correction. Note that the pinhole model is able to calibrate all the rays, while the unconstrained model can be populated only by the rays that hit the scanned surface, thus they are a bit less. Also note that all the miscalibrated rays have (apparently) disappeared after outlier removal.

jector. While this could seem to be a limitation, it should be noted that our approach is designed for a very specific, yet critical, scenario. Namely, our goal is to augment the coverage of standard 3D scanners that are designed to use a pair of stereo cameras coupled with non-calibrated projector, which are very common among commercial products. To validate our method, we will perform specific experiments to show that it is indeed able to significantly increase the number of acquired points. This happens without sacrificing quality and without any modification for off-the-shelf systems.

7.2 High-Coverage 3D Scanning

The main idea of our approach is to exploit the 3D points triangulated by two calibrated cameras to get insight about the projector geometry. Basically, this happens by collecting among subsequent shots the coordinates 3D points that reproject exactly over the same projector pixel and then using a simple least square fitting to fix the parameters of the projector ray associated to that pixel. In order to easily perform this step and to cope well with commercial quality projector optics, we adopted the general unconstrained camera model. In Chapter 6 we already studied a method for effectively calibrating such model and now we are extending it to deal with this new application.

7.2.1 Online Projector Calibration

Our goal is to calibrate the rays of the projector as if it was a camera that always recaptures the exact pattern that is projected. This assumption, which is similar to many other projector calibration methods, allows to implicitly know exactly the observed codes \mathbf{Co} . By contrast, the expected code \mathbf{Ce} must be obtained from the scene. Most approaches found in literature solve this problem by projecting some pattern on a known target and using the features of the target as references. In our case, since we

have two calibrated cameras available, we can obtain the 3D reference points by collecting them during the scanning process.

The triangulation of each point of the scene happens by finding the same code (computed from the sequence of patterns [118]) on both cameras. This will produce 3D points that are associated to codes observed by camera pixels, that in general do not correspond to the expected code \mathbf{C}_e for any projector ray. For this reason, we cannot directly use the 3D points belonging to the acquired surface, rather we must produce additional points corresponding exactly to the (virtually) observed codes. This can be done easily by interpolating the camera rays whose codes encompass the observed code \mathbf{C}_o with weights inversely proportional to the distance from \mathbf{C}_o of their respective measured codes, as described previously in Section 6.4.

With this interpolation strategy we make no assumption on the topological structure of the rays thus the only way to define a notion of nearest neighbours is through the observed codes itself. However, if we assume the ordering induced by the lattice topology of the CCD, we can reason on an interpolation function that uses the 4 neighbouring rays strictly containing the interpolated code. The four codes $\mathbf{C}_{o_1} \dots \mathbf{C}_{o_4}$ observed by the 4 nearest rays form a quadrilateral in the two-dimensional code space. Consequently, they can be remapped through an homography \mathbf{H} to the normalized space given by the versors $(1, 0)^T$ and $(0, 1)^T$. At this point, the expected code \mathbf{C}_e can be itself mapped to the normalized space through \mathbf{H} producing a normalized expected code \mathbf{C}_{e_n} that, by construction, will be contained into the square S given by the vertices $(0, 0)$, $(1, 0)$, $(1, 1)$, $(0, 1)$. Finally, the interpolation weights associated to each of the four rays are the same ones that would be used for a bilinear interpolation inside S .

Despite the specific interpolation data and weights used, two virtual rays are obtained from the camera stereo pair. These newly obtained virtual rays can finally be used to triangulate the expected point \mathbf{C}_e corresponding to \mathbf{C}_o and use it to perform the same optimization described in Chapter 6. Note, however, that equation (6.7) holds only for locally planar surfaces. Generally speaking, this is not guaranteed if the points \mathbf{C}_e are obtained from the scanning of random objects. Still, since both man-made and natural objects usually exhibit several low frequency areas, it is reasonable to guess that at least a sizeable portion of the obtained expected points will be accurate enough. Moreover, the scanned object will likely move during different shots, enhancing the coverage of the projector frustum and eventually adding redundancy. Finally, even if some \mathbf{C}_e could suffer from bad estimation, in the next section we suggest an apt method to filter outliers.

7.2.2 Outliers Filtering

After estimating the projector rays, we can assess their quality by means of their fitting residual. Specifically, we can set a badness threshold that can be used to deem as unreliable rays that obtain a bigger residual. Such rays can be removed, in which case they will be simply unavailable for triangulation (note that there is no need to estimate all of the projector rays). Otherwise, it is indeed possible to still recover them by filter-

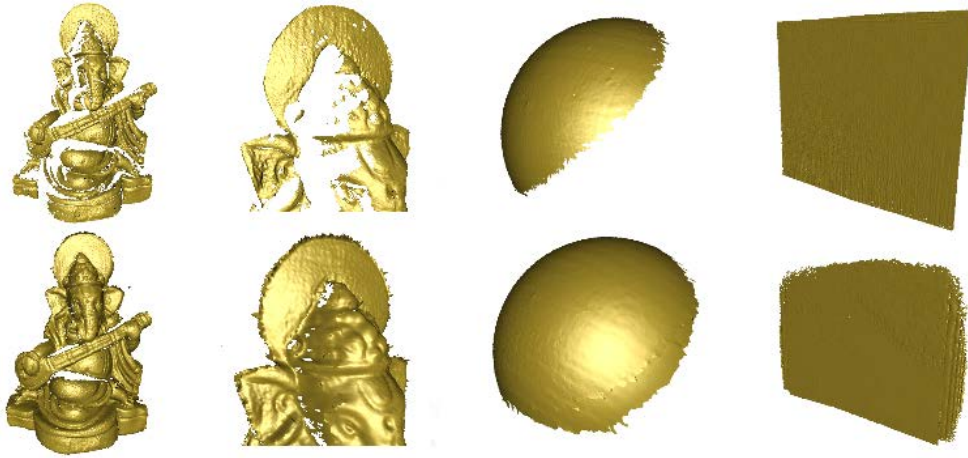


Figure 7.3: Coverage difference between the baseline (top row) and the unconstrained method (bottom row) for some different subjects.

ing inaccurate \mathbf{C}_e points that could possibly be the cause of the bad estimation. To do this, we create a tentative ray candidate by interpolating its available neighbours. Afterwards, this candidate is used to gather associated \mathbf{C}_e that are near a given threshold to it, which in turn can be used to obtain a new estimate for the original ray. The rationale of this method is that the interpolation of the neighbours (if any) would result in a putative ray good enough for an effective inlier selection. In Fig. 7.2 we show the bundles of projector rays obtained after calibration. The first image depicts the bundle obtained by calibrating the projector with a standard pinhole model. This happens by using the estimated \mathbf{C}_e as the 3D points of a virtual calibration objects and the associated projector codes as their reprojections. While this could seem a good approximation, we will show in the experimental section that the pinhole model is not able to fully deal with the imperfection of commercial quality lenses (as also observed in Chap. 6). The other two show the bundles obtained using the described unconstrained model respectively before and after outlier filtering and ray correction.

7.3 Experimental Evaluation

In order to evaluate the proposed method we built an experimental setup similar to many off-the-shelf structured light 3D scanners (see Fig. 7.4). We accurately calibrated the two 1280x1024 pixels cameras for both intrinsic and extrinsic parameters according to both the pinhole model and to the unconstrained model. The projector used is an SVGA Dlp micro projector. We implemented three reconstruction models:

- *Baseline*: the unconstrained stereo camera reconstruction model that works without needing projector calibration presented in Chapter 6. We expect this to be the most accurate but to exhibit less coverage;



Figure 7.4: Scanner head with 2 calibrated cameras and an uncalibrated projector.

- *Pinhole*: a reconstruction configuration that uses the projector calibrated according to the pinhole model (including distortion) to enhance coverage. We expect this to be less accurate due to the limitation of the pinhole model, especially for the cheap optics of commercial-quality projectors;
- *Unconstrained*: the reconstruction model using the unconstrained projector calibrated with the approach described in this chapter.

We tested these models by scanning three different objects: a small figurine of Ganesha, which exhibits small details and thus many high frequency areas, a regular sphere, which is rather smooth and includes only low frequencies, and finally a flat plane, used as a featureless reference object. For each object we acquired about 100 scans covering almost all the projector frustum, and we compared the results obtained by calibrating the projector with different amounts of randomly selected shots subsets. We adopted four different evaluation criteria that are described in the following subsections.

7.3.1 Enhanced Coverage

With this test we measure the ratio between the area of the surface obtained with the calibrated projector methods and with baseline. This metric represents the enhancement in terms of coverage. In Fig. 7.5 we plot the coverage increment with respect to the number of shots used to calibrate (which correspond to the total number of scans, since the method is online). Of course we show only the curves for the Sphere and

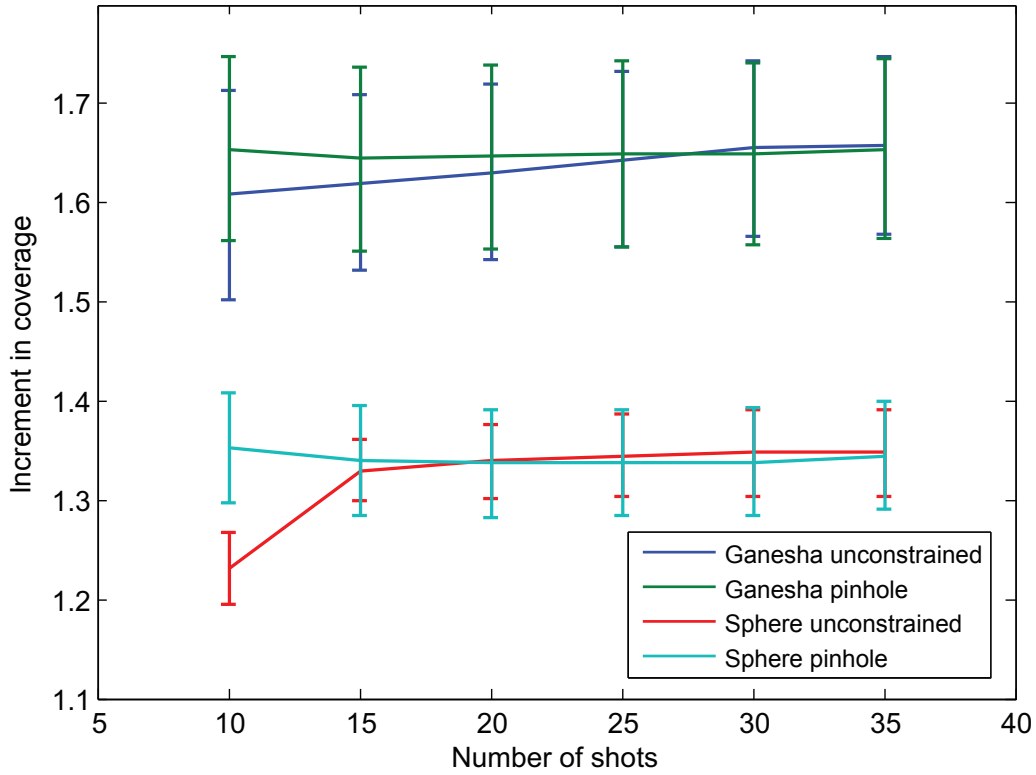


Figure 7.5: Increment in the coverage with respect to the number of scans.

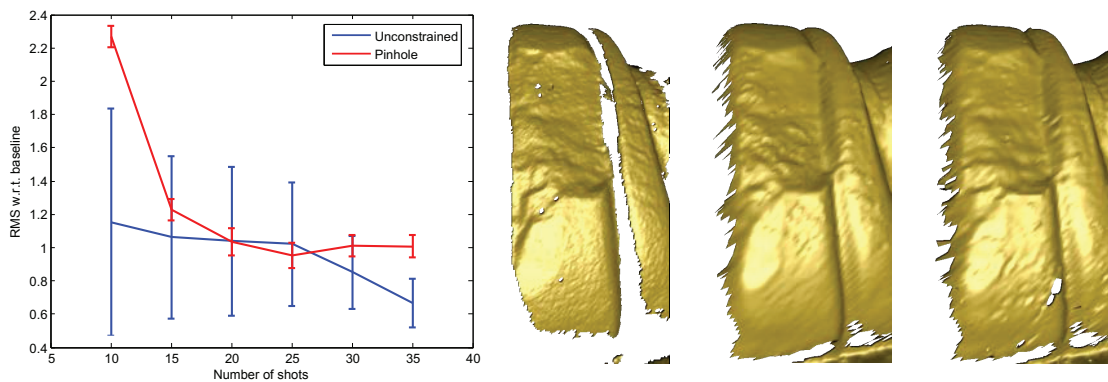


Figure 7.6: Accuracy of the reconstruction with respect to the baseline method. The close-ups on the left part of the figure show a detail of the reconstruction obtained respectively with the baseline, unconstrained and pinhole methods.

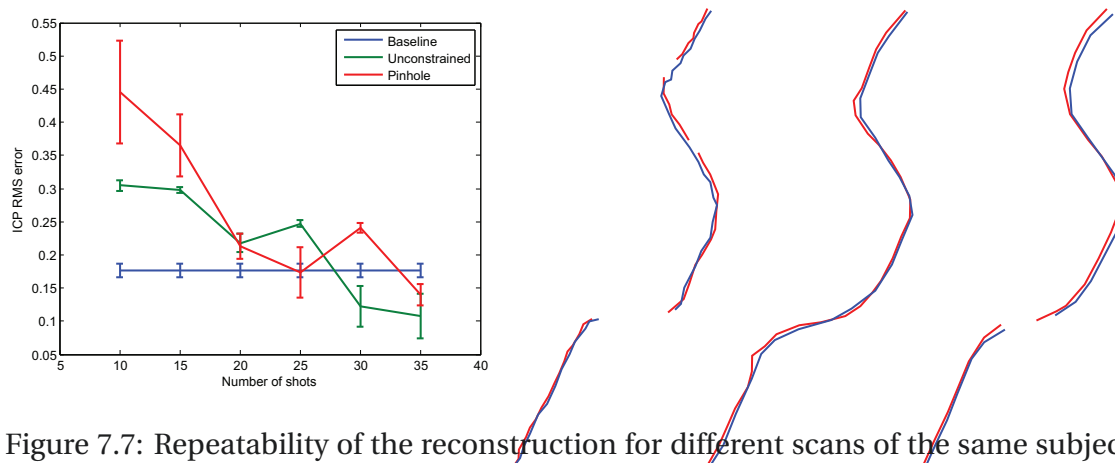


Figure 7.7: Repeatability of the reconstruction for different scans of the same subject. On the right part of the figure we show some slices from the acquired meshes to illustrate the alignment between subsequent scans respectively with the baseline, pinhole and unconstrained methods.

Ganesha objects, since there is no increment for the plane (which is equally well seen by both cameras). Note that the latter object obtains a larger advantage since it contains many convoluted areas that are hard to capture at the same time by two cameras and the projector. Note also that the pinhole model reaches immediately the maximum increment while the unconstrained model requires from 15 to 30 shots to perform equally well. This is expected since in this case the calibration includes all the rays from the start. However, we will see in the next test that this advantage comes at the cost of a lower accuracy. Some qualitative examples of the coverage are shown in Fig. 7.3. Here the scattered edges of the plane are due to the fact that not all the projector rays have been recovered. This happens simply because the rays on the periphery of the frustum appears in fewer scans of the subject, which is expected. If a full coverage of the projection must be guaranteed, this can be obtained offline using a bigger planar object encompassing the whole frustum.

7.3.2 Reconstruction accuracy

To give a reasonable accuracy measure, we decided to adopt the baseline method as a reference. This is a reasonable choice since we already discussed the accuracy of a camera pair calibrated with the unconstrained model. In this regard, we express the accuracy as the RMS error, after ICP registration [33], between the acquired surface and the "ground truth" offered by the baseline. Note that such RMS is expressed in world unit, which, since the cameras have been calibrated with a computer monitor, corresponds to the size of a pixel on that specific screen (approximately 0.2 mm). In Fig. 7.6 we show the obtained accuracy after different amounts of scans. The pinhole method requires few shots to reach its maximum accuracy. However it always performs worse than the unconstrained method. Furthermore the standard deviation of the pinhole curve is narrower. These phenomena can be explained respectively by the fact that the

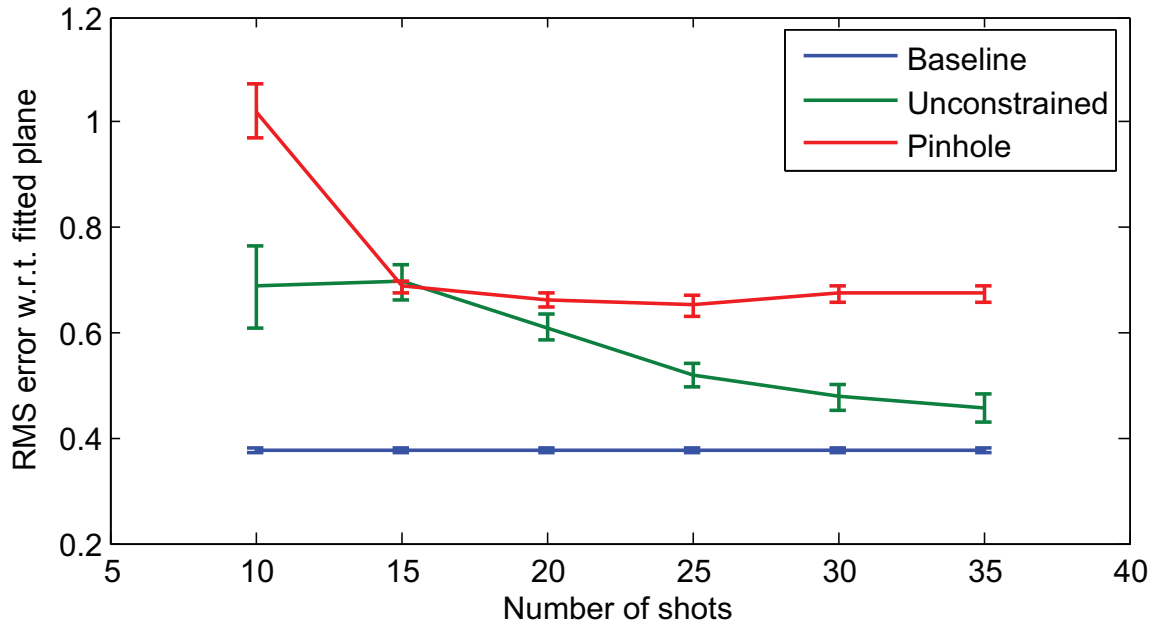


Figure 7.8: Coherence of the reconstructed surface with a planar reference target.

pinhole model is not able to fully handle the imperfections of a real lens and that its statistical nature makes it very stable with respect to the set of shots selected for calibration. The unconstrained method, albeit after several shots, allows for a significantly better accuracy.

7.3.3 Surface Repeatability

While the accuracy measures the compliance of the results with respect to the ground truth, we are also interested in the repeatability of the reconstruction within the same method. To evaluate this metric we took several scans of the same subject with slightly different poses and we computed the average RMS error, after ICP registration, between the surfaced acquired using the same method. Basically, this measure gives us an insight about the resilience of the method to random noise and to aliasing error generated by the interplay between camera and projector rays. In Fig. 7.7 we plot such measure for the baseline method (which appears as a horizontal line since it does not depends on the number of scans) and of the other two methods. We can conclude that all the tested approaches exhibit a good repeatability, in the order of hundredths of a millimetre. This repeatability appears to be not strongly sensitive to the number of scan used to calibrate, with the possible exception of the pinhole method that performs less well with few shots.

7.3.4 Planarity

Finally, we measured the planarity of the reconstruction of a reference plane made by coating with matte paint a float glass. This is done by computing the average RMS error with respect to a general plane that has been fitted to the data. The rationale of this measure is to assess spatial distortions that usually characterizes imperfect calibrations. The results obtained are shown in Fig. 7.8. We can observe that the pinhole method produces the surface with larger distortion. This is certainly attributable to the inherently imperfect correction of the distortion.

7.4 Conclusion

In this chapter we introduced an online projector calibration method based on the unconstrained imaging model that can be seamlessly applied to many commonly available 3D scanning systems. The main advantage of this method is that it can be performed during the normal scanning process, allowing an improved scene coverage with little or no additional effort. Furthermore, we have shown by extensive experiments that an increase ranging from 30 to 60 percent in the recovered surface area can be easily obtained without sacrificing the reconstruction accuracy.

8

Non-Parametric Lens Distortion Estimation for Pinhole Cameras

In this chapter we propose a raxel-based camera model where each imaging ray is constrained to a common optical center, thus forcing the camera to be central. Such model can be easily calibrated with a practical procedure which provides a convenient (model-free) undistortion map that can be used to obtain a virtual pinhole camera. The proposed method is so general that can also be adopted to calibrate a stereo rig with a displacement map that simultaneously provides stereo rectification and corrects lens distortion.

8.1 Introduction

Given the high variability of non-pinhole cameras, it has been proven to be very difficult to define a parametric distortion model able to accommodate the diverse behaviour of physical lenses. This hindrance has been addressed by introducing general camera models based on unconstrained rays (usually called raxels) [80, 169] as well as non parametric (albeit still radial) distortion models [84, 174].

Traditionally, the literature has deemed the unconstrained models and related calibration procedures a last resort to be adopted only when traditional approaches fail due to either geometrical or methodological issues. This is due to the fact that their flexibility comes at the price of a higher calibration complexity and (sometimes) cruder approximations. Recent research, however, shows that, with the aid of a dense coded target, a fully unconstrained imaging model can be applied effectively to real-world pinhole cameras obtaining better accuracy and without needing a complex calibration procedure [32]. However, the advantages in terms of precision, are partially offset by the inability to use the wide mathematical toolset devised for the pinhole model. This is unfortunate, since when dealing with most standard cameras without extreme distortions, the central model is still reasonable.

With this study, we try to fill the gap between traditional pinhole calibration techniques and unconstrained models. Namely, we propose a model where the only constraint applied to raxels is that they are required to cross a common projection center. Under this assumption, after performing a proper calibration, it is very easy to define a non-parametric displacement map over the image plane that can be used to move back and forth from the unconstrained model to a virtual pinhole camera. This, in turn, allows to exploit the full arsenal of tools designed to work with pinhole cameras. To this end, the contribution of this work is threefold. First, we introduce an effective and practical calibration procedure and optimization schema for the proposed semi-constrained model. Second, we define an optimal approach to create a virtual pinhole camera from the calibrated model. Finally, we show how to naturally extend the method to the calibration and rectification of a stereo pair. The accuracy and effectiveness of our proposal is experimentally assessed and compared with both the fully unconstrained camera and the current state-of-the-art parametric distortion models.

8.2 Unconstrained Distortion Map for Central Camera Calibration

To estimate a dense non-parametric lens distortion we start by recovering the 3D light rays associated to each image pixel. Specifically, we formalize the light path entering the lens and hitting the sensor at discrete pixel coordinate (u, v) with the straight line $r_{(u,v)} = (o, d_{(u,v)})$ passing through the point o and oriented along the direction $d_{(u,v)}$. The common point o constrains our model to be central while no restriction is enforced on the directions $d_{(u,v)}$. Also, the uniform-spaced grid of the CCD provides an ordering on

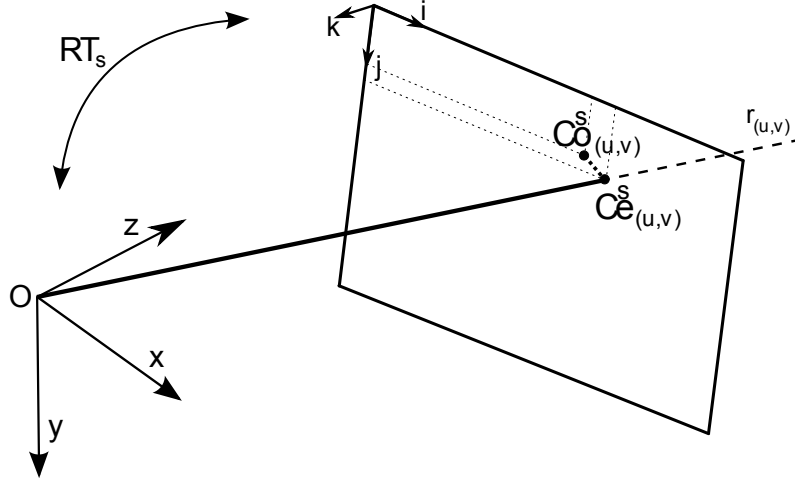


Figure 8.1: Schema of the optimized camera model involving the optical center o , the ray direction, the observed and expected code and a target pose

the rays spatial topology.

Since the model implies 3 degrees of freedom for each pixel, plus additional 3 for the optical center o , standard point-to-point calibration approaches like [206, 179, 55] cannot provide enough data to recover a valid solution. We solve this by adopting the dense structured-light target proposed in [32]. Specifically, we use an high-resolution LCD screen displaying phase shift patterns to uniquely localize each target point seen by the camera. This has a strong advantage with respect to common calibration targets composed by a discrete set of features: unlike methods based on corner or ellipse localization, we can reason in terms of discrete camera coordinates. Indeed, for each camera pixel (u, v) a precise sub-pixel localization of the 2D target-space coordinate of the observed point can be recovered from the phase unwrapping process.

To estimate the optical center and the direction of each ray, the calibration target is exposed to the camera in different positions and orientations. We denote with \mathbf{RT}_s the 3×4 matrix describing the roto-translation of the target with respect to the camera in the pose s , assuming the target reference frame located in the upper left monitor pixel with $\vec{i}, \vec{j}, \vec{k}$ versors being respectively the monitor columns, rows and normal. For each pose, let $\mathbf{Co}_{(u,v)}^s \in \mathbb{R}^2$ be the code observed by the ray (u, v) in shot s . Since the LCD geometry and the displayed codes are known, the intersection between a ray $r_{(u,v)}$ and the target plane defined by a pose \mathbf{RT}_s yields to the expected code $\mathbf{Ce}(r_{(u,v)}, \mathbf{RT}_s) \in \mathbb{R}^2$ that the ray should have observed (Fig 8.1).

8.2.1 Single Camera calibration

Following [32], we recover the geometry of the rays entering the camera as the generalized least-squares minimization problem:

$$\operatorname{argmin}_{\mathbf{r}(u,v), \mathbf{RT}_s} \sum_{u,v,s} (\boldsymbol{\varepsilon}_{(u,v)}^s)^T (\boldsymbol{\Sigma}_{(u,v)}^s)^{-1} \boldsymbol{\varepsilon}_{(u,v)}^s \quad (8.1)$$

where $\boldsymbol{\varepsilon}_{(u,v)}^s = \mathbf{Co}_{(u,v)}^s - \mathbf{Ce}(r(u,v), \mathbf{RT}_s)$ are the residuals between the observed and expected codes and $(\boldsymbol{\Sigma}_{(u,v)}^s)^{-1}$ is the error covariance matrix for the given ray-pose combination.

In our setting, we aim to simultaneously minimize the optical center o , the direction $d_{(u,v)}$ of all rays and the pose RT_s for each exposure of the target. Similarly to [32], we can also take advantage of the conditional independence of the parameters to implement an alternating optimization scheme that seeks optimal o and $d_{(u,v)}$ assuming last estimation of RT_s fixed, and vice-versa. While our optimization involves less parameters, the optimization itself is more complex since the common optical center introduces a coupling between the rays which cannot be estimated independently anymore. As a consequence, the rays optimization step simultaneously estimates the optical center o and the ray directions $d_{(u,v)}$ for each pose set, while we adopt the same ICP-based poses estimation step introduced in [32]. The former step is discussed in detail in section 8.2.1 while, for the latter, we refer the reader to the original paper.

To start the alternating optimization we need a good initial approximation for the involved parameters. To provide such data, we gather a set of 3D-2D point correspondences assuming a discrete grid of target points similar to what can be commonly obtained with a chessboard. Then, we use *calibrateCamera* function provided by OpenCV [36] to obtain target poses for each exposure and the direction of each ray.

Optical Center and Rays Direction Optimization

In the o and $d_{(u,v)}$ optimization step we consider target poses constant. Let

$$\mathbf{x}_{(u,v)}^s = RT_s \begin{pmatrix} \mathbf{Co}_{(u,v)}^s \\ 0 \\ 1 \end{pmatrix}$$

be the 3D coordinates of the observed code $\mathbf{Co}_{(u,v)}^s$ transformed through the pose RT_s . Since the generalized least squares formulation with respect to the target coordinates corresponds to a linear least squares with the distance of each ray and its associated 3D point $\mathbf{x}_{(u,v)}^s$ ¹, we can formulate the estimation of the optical center o as following:

$$\operatorname{argmin}_o \sum_{u,v} \min_{d_{(u,v)}} \sum_s \| (h_{(u,v)}^s)^T (I - d_{(u,v)} d_{(u,v)}^T) \|^2 \quad (8.2)$$

where $h_{(u,v)}^s = (\mathbf{x}_{(u,v)}^s - o)$. We start by re-writing the squared norm in (8.2) as $(h_{(u,v)}^s)^T (I - d_{(u,v)} d_{(u,v)}^T) h_{(u,v)}^s$ to obtain

¹For a complete proof see [32]

$$\operatorname{argmin}_o \sum_{u,v} \sum_s \|h_{(u,v)}^s\|^2 - \max_{d_{(u,v)}} \sum_s (d_{(u,v)}^T h_{(u,v)}^s)^2 \quad (8.3)$$

Let $\bar{x}_{(u,v)}$ be the centroid of the point cloud generated by the intersections of the ray (u, v) and the target for each observed pose. Also, let $\bar{h}_{(u,v)} = (\bar{x}_{(u,v)} - o)$ be the distance vector between o with such centroid. By expressing $h_{(u,v)}^s$ as the summation of the two components:

$$h_{(u,v)}^s = (\mathbf{x}_{(u,v)}^s - \bar{x}_{(u,v)}) + \bar{h}_{(u,v)}$$

and expanding the formulation in (8.3) we obtain:

$$\operatorname{argmin}_o \sum_{u,v} N_{(u,v)} (tr(\mathbf{S}_{(u,v)}) + \|\bar{h}_{(u,v)}\|^2) - \quad (8.4)$$

$$- \max_{d_{(u,v)}} d_{(u,v)}^T (N_{(u,v)} \mathbf{S}_{(u,v)} + N_{(u,v)} \bar{h}_{(u,v)} \bar{h}_{(u,v)}^T) d_{(u,v)} \quad (8.5)$$

where $\mathbf{S}_{(u,v)}$ and $N_{(u,v)}$ are respectively the covariance matrix and the cardinality of the point cloud generated by $r_{(u,v)}$.

Since we start our optimization with a configuration close to the optimum, we expect that the distance between each ray and its expected code is as small as few target pixels. This implies that the spatial extent of each point cloud is order of magnitude smaller than the distance $\bar{h}_{(u,v)}$. Under this assumption, an approximate maximizer for (8.5) is given by

$$d_{(u,v)} = \frac{\bar{h}_{(u,v)}}{\|\bar{h}_{(u,v)}\|^2} \quad (8.6)$$

By substituting (8.6) into (8.4) and (8.5), after some simplifications, we obtain the following alternative formulation

$$\operatorname{argmax}_o \sum_{u,v} N_{(u,v)} \frac{\bar{h}_{(u,v)}^T \mathbf{S}_{(u,v)} \bar{h}_{(u,v)}}{\|\bar{h}_{(u,v)}\|^2} \quad (8.7)$$

Problem (8.7) cannot be solved in a closed form. To provide a good approximate solution we compute the derivative with respect to o :

$$\begin{aligned} \frac{\partial}{\partial o} \sum_{u,v} N_{(u,v)} \frac{\bar{h}_{(u,v)}^T \mathbf{S}_{(u,v)} \bar{h}_{(u,v)}}{\|\bar{h}_{(u,v)}\|^2} \\ = \sum_{u,v} 2N_{(u,v)} K_{(u,v)} \bar{h}_{(u,v)} \end{aligned} \quad (8.8)$$

$$K_{(u,v)} = \frac{(-\mathbf{S}_{(u,v)} \|\bar{h}_{(u,v)}\|^2 + \mathbf{I}(\bar{h}_{(u,v)}^T \mathbf{S}_{(u,v)} \bar{h}_{(u,v)}))}{\|\bar{h}_{(u,v)}\|^4} \quad (8.9)$$

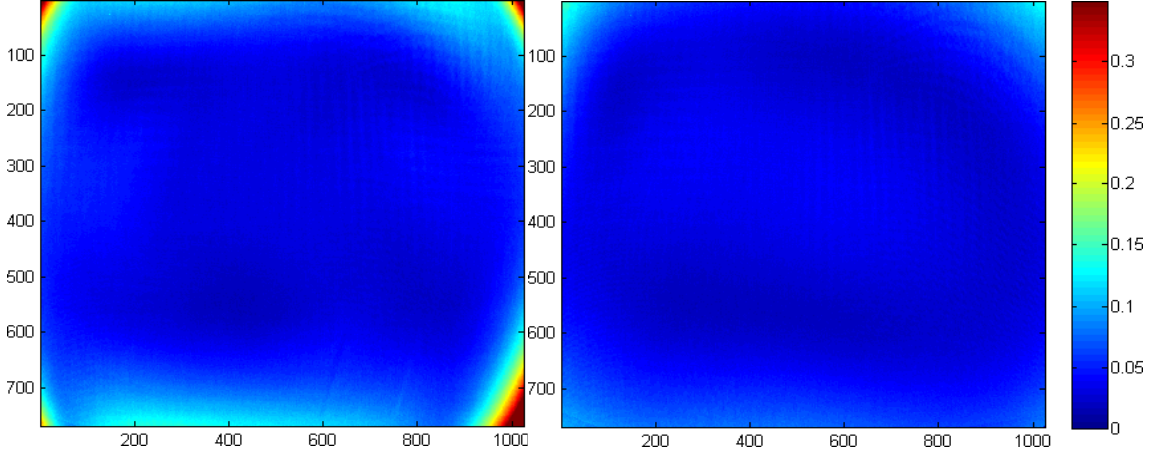


Figure 8.2: RMS of the error between the observed and the expected code for each $r_{(u,v)}$ at the first (left image) and last (right image) iteration of the optimization process of rays, optical center, and poses.

If $K_{(u,v)}$ is known, o can be obtained by setting to zero Equation (8.8) and solving the resulting linear system:

$$\sum_{u,v} 2N_{(u,v)} K_{(u,v)} o = \sum_{u,v} 2N_{(u,v)} K_{(u,v)} \bar{x}_{(u,v)} \quad (8.10)$$

Since $K_{(u,v)}$ is itself a function of o , the maximization problem (8.7) is tackled iteratively by computing $K_{(u,v)}$ with the estimate of o at iteration $t-1$ and then solving (8.10) to obtain a new estimate at iteration t and repeating this process until $\|o^{(t)} - o^{(t-1)}\| < \epsilon$.

When the optical center is found, the direction of each ray is computed with equation (8.6). A qualitative result of the effect of the optimization process is shown in Figure 8.2.

From ray bundle to virtual pinhole camera

After the optimization of rays, optical center and poses we obtain a detailed model describing the light path entering the camera. Next, we need to choose a convenient image plane that define the intrinsic parameters of a new virtual pinhole camera, along with a non-parametric dense undistortion function to be applied to the acquired images.

As a preliminary step, all rays are translated so that their unique intersection point o lies at the origin. After that, we define a plane φ as the locus of points x satisfying $\langle x - v_\varphi, n_\varphi \rangle = 0$, with $n_\varphi = \frac{v_\varphi}{\|v_\varphi\|}$. As soon as an image plane φ is chosen, it generates a virtual camera with the z-axis oriented as the vector n_φ and with a focal length $f = \|v_\varphi\|$. When choosing any φ intersecting the ray bundle, all the intersection points inherit the lattice topology from the camera sensor. By re-sampling this lattice in a

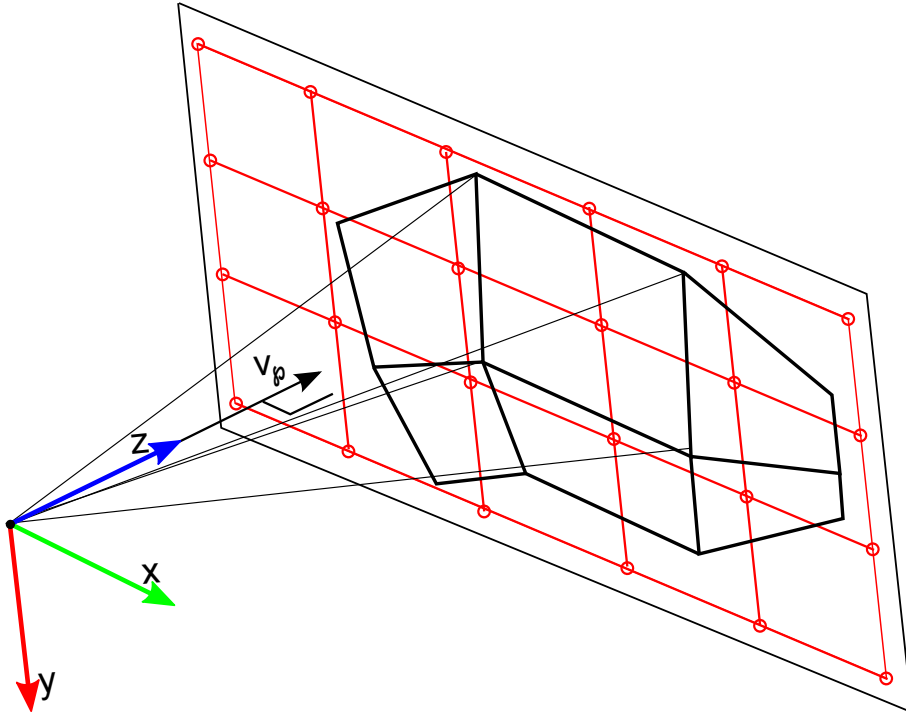


Figure 8.3: Spatial configuration of the optimized camera. Camera z-axis is aligned with the plane v_φ . The intersection between all rays and the plane inherits the lattice topology of the camera sensor. The points lattice (in black) on the image plane is resampled in a uniform grid (in red) to create the undistortion function.

uniform grid, a new undistorted image can be created. Additionally, the grid position and size define the projection of the optical center on the image plane (i.e. the pinhole parameters c_x and c_y) and the undistorted image size (Fig. 8.3).

Image Plane Estimation

Virtually any plane φ intersecting the ray bundle can be used to generate a valid new virtual camera. However, the shape of intersections lattice strongly depends on the plane normal n_φ whereas its size is proportional to the plane distance from the origin $\|v_\varphi\|$. As a consequence, choosing a good plane is crucial to ensure the lattice be as regular as possible and reduce the approximation error made by the subsequent resampling and interpolation processes.

We tackle these two problems in two subsequent steps. First, we estimate the plane orientation (i.e. assuming an unitary distance from the origin) to minimize the variance of the squared distance between each plane-ray intersection point and its neighbours. This ensure the lattice to be as regularly shaped as possible. After that, the scaling factor (i.e. the distance of the plane) is computed so that the average distance

between all the points is equal to 1.

Let $I_d \in \mathbb{R}^2$ be the set of all valid (u, v) indices of the rays. Let $U(i \in I_d) = U(u, v) = \{(u-1, v), (u+1, v), (u, v-1), (u, v+1)\}$ the function defining the set of four neighbours of a ray indexed by i . The squared distance between the 3D intersections generated by two rays r_i and $r_{j \in U(i)}$ with a plane φ lying at unitary distance from the origin is given by:

$$D_{i,j}^2 = \left\| \frac{d_i}{n_\varphi^T d_i} - \frac{d_j}{n_\varphi^T d_j} \right\|^2 \quad (8.11)$$

Consequently, the variance of the squared distances $D_{i,j}^2$ between each ray and its neighbours is given by the function

$$f_D = \sum_i \sum_{j \in U(i)} \left(D_{i,j}^2 \right)^2 - \left(\sum_i \sum_{j \in U(i)} D_{i,j}^2 \right)^2 \quad (8.12)$$

We cast the plane orientation problem as the minimization

$$\begin{aligned} \underset{n_\varphi}{\operatorname{argmin}} \quad & f_D \\ \text{subject to} \quad & \|n_\varphi\| = 1 \end{aligned} \quad (8.13)$$

solved via geodesic steepest descent. We start with an initial estimate of $n_\varphi^{(0)} = (0 \ 0 \ 1)^T$. For each iteration t , we update the estimate of $n_\varphi^{(t)}$ enforcing the constraint of $\|n_\varphi\| = 1$ by rotating $n_\varphi^{(t)}$ around the rotation axis $\Psi = \nabla f_d^{(t-1)} \times n_\varphi^{(t-1)}$ for an angle $\theta = \lambda \min(\|\Psi\|, \epsilon)$. The constant λ affects the speed of the gradient descent while ϵ gives an upper bound on the amount of rotation to avoid instabilities.

To perform effectively the optimization, ∇f_d can be analytically computed as follows:

$$\nabla f_d = \sum_i \sum_{j \in U(i)} 2D_{i,j}^2 \frac{\partial}{\partial n_\varphi} D_{i,j}^2 - \quad (8.14)$$

$$- \left(\sum_i \sum_{j \in U(i)} D_{i,j}^2 \right) \left(\sum_i \sum_{j \in U(i)} \frac{\partial}{\partial n_\varphi} D_{i,j}^2 \right)$$

$$\frac{\partial}{\partial n_\varphi} D_{i,j}^2 = \frac{2}{(n_\varphi^T d_i)^2} \left(\frac{d_i^T d_j}{n_\varphi^T d_j} - \frac{d_i^T d_i}{n_\varphi^T d_i} \right) d_i + \quad (8.15)$$

$$+ \frac{2}{(n_\varphi^T d_j)^2} \left(\frac{d_j^T d_i}{n_\varphi^T d_i} - \frac{d_j^T d_j}{n_\varphi^T d_j} \right) d_j$$

Generating the Undistortion Map

Once an optimal plane has been found, we setup an interpolation procedure to re-sample the points on a regular grid. Let $p_{(u,v)}$ be the intersection of $r_{(u,v)}$ with the

optimized plane φ . First, all the points $p_{(u,v)}$ are rotated around the origin so that the principal point v_φ coincides with the z-axis. After discarding the third component of all the points (all equal to 1 after the rotation), we compute the integral coordinates of the top-left $tl_p \in \mathbb{Z}^2$ and bottom right $br_p \in \mathbb{Z}^2$ corners of the containing bounding-box. After this step, we can provide the intrinsic matrix of the new virtual pinhole camera as

$$K = \left(\begin{array}{cc|c} \|v_\varphi\| & 0 & -tl_p \\ 0 & \|v_\varphi\| & \\ \hline 0 & 0 & 1 \end{array} \right) \quad (8.16)$$

The undistorted image associated to the camera described by K corresponds to a unit-spaced grid inside the area of the bounding box. This leads to the construction of a dense displacement function $U_d : B \rightarrow \mathbb{R}^2$ that maps the coordinates of the output undistorted image $B \subset \mathbb{N}^2$ to sub-pixel coordinates of the input image².

To produce the displacement map U_d , we generate the quadrilaterals $q_1 \dots q_n$ formed when considering the 4-neighbours connectivity of the points $p_{(u,v)}$ with the topology induced by the rays lattice. For each quadrilateral q_i , we compute the homography H_i that transform the inner space bounded by its four vertices into the square defined by the CCD location of the four rays associated to each vertex. Then, the displacement map U_d can be simply obtained by:

$$U_d(u', v') = H_{Q(u', v')}(u', v', 1)^T \quad (8.17)$$

where $Q(u', v')$ is the function that returns the index of the quadrilateral containing the point $(u' \ v')^T$, if exists.

Filtering data outliers

Apart for being central, our model gives no constraint on the relative position of the rays. As a consequence, erroneous data caused by failures in the phase decoding process may lead to outliers in the ray estimation. Since no regularization is involved, we included a data filtering step at each alternation of rays-poses optimization. Specifically, we define the function $E(u, v)_s : I_d \rightarrow \mathbb{R}$ as the point-line distance between the ray $r_{(u,v)}$ and the point $x_{(u,v)}^s$. We then filter the observed codes $\mathbf{Co}_{(u,v)}^s$ by considering the median of the error function E in a squared neighbourhood of each point (u, v) . If $E(u, v)_s$ is greater than κ times the median, $\mathbf{Co}_{(u,v)}^s$ is marked as invalid and not used any more in the subsequent iterations. Rays with less than 5 observed codes are completely removed from the optimization. A qualitative example of the output of the filtering process is shown in Figure 8.4.

Even if we filter erroneous observed codes, it may happen to obtain some quadrilaterals q_i for which the topological order of the vertices is not coherent with the order of the relative rays. Since this would lead to a non-injective displacement map U_d , we

²The size of the produced undistorted image is equal to the size of the bounding box

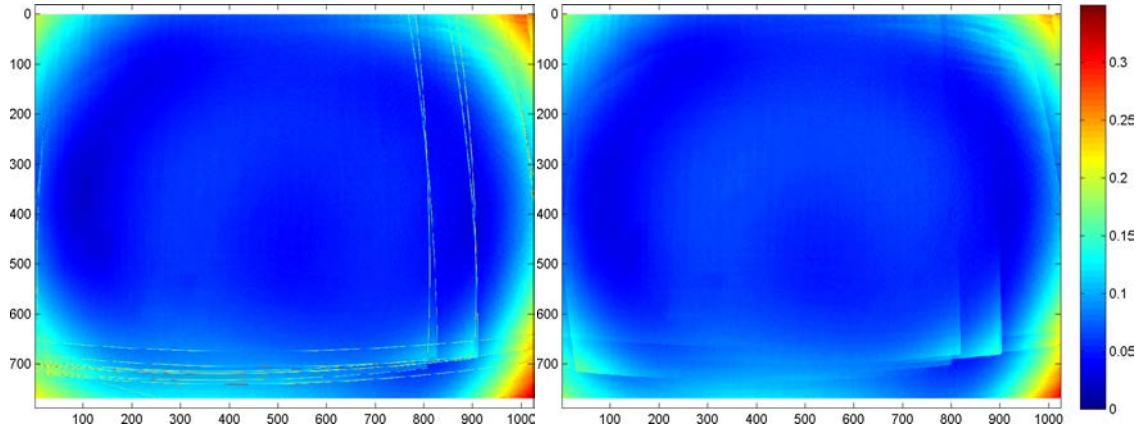


Figure 8.4: The effect of the observed codes filtering step displayed by accumulating the value $E(u, v)$ among all the poses s . Left: original data may contain clear outliers near the boundary of the target. Right: after the filter almost all the spikes are no longer visible.

selectively replace the rays associated to that quadrilateral with a linear interpolation of the neighbours.

8.2.2 Dealing with Stereo Cameras

Since each ray acts independently with respect to the others, our approach can be easily extended to simultaneously calibrate and rectify a stereo rig. The pose optimization step remains exactly the same with the foresight to merge the two bundle of rays associated to each camera. Conversely, the optical centre and rays direction optimization can be performed independently on the two sets operating the same instance of target poses.

As a starting configuration for the subsequent optimization we performed the intrinsic and extrinsic calibration of the camera rig using the function provided by OpenCV library. Then, we consider the two cameras as one by moving the rays originating by the right camera to the reference frame of the other. At the end of the optimization, we obtain an estimate of the two optical centres o_1 and o_2 and the directions of the rays in the two bundles. From this point, we roto-translate the rays to let o_1 coincide with the origin and the epipole $e = (o_2 - o_1)$ being oriented along the x-axis.

Rectification and Undistortion Map

If we constrain the image plane optimization so that n_φ remains orthogonal to e , the estimated plane would have the property to keep all the epipolar lines for the left and right cameras being parallel. To achieve this, we slightly modify the optimization discussed in section 8.2.1 by fixing the rotation axis $\Psi = \frac{e}{\|e\|}$ and the rotation angle to

$$\theta = \lambda \min(\langle \nabla f_d^{(t-1)}, \Psi \times n_\varphi^{(t-1)} \rangle, \epsilon).$$

After image plane optimization, two sets of points are generated by the intersection of the two ray bundles with the plane. The set of points generated by the right camera is translated in the opposite direction of the x-axis by the length of the baseline $T = \|e\|$ to let the right optical center coincide with the left one. Subsequently, two different bounding boxes are generated with the two sets of points. The height of the two boxes (i.e. the vertical coordinates of the top-left and bottom-right corners) are forced to be equal so that the epipolar lines are coincident with the rows of the two images. To keep the largest available common area between the two images, the left edge of the merged bounding box is taken from the bounding box of the right point set. Symmetrically, the right edge is taken from the left point set. Note that, this way, the intrinsic matrices of the two cameras are exactly the same.

Finally, we compute the reprojection matrix

$$Q = \left(\begin{array}{ccc|c} 1 & 0 & 0 & -tl_p \\ 0 & 1 & 0 & \\ \hline 0 & 0 & 1 & \|v_\varphi\| \\ 0 & 0 & 1/T & 0 \end{array} \right) \quad (8.18)$$

so that, given any dimensional image point (u', v') and its associated disparity d , it can be projected into four-dimensional projective space with

$$\begin{pmatrix} x \\ y \\ z \\ w \end{pmatrix} = Q \begin{pmatrix} u' \\ v' \\ d \\ 1 \end{pmatrix} \quad (8.19)$$

8.3 Experimental Section

In order to assess the performance of the proposed approach, we compared it against the unconstrained model [32] and the rational distortion model proposed by Claus and Fitzgibbon [55] in both single camera and stereo setups.

Our test setup included two PointGrey Flea3 1Mp grayscale cameras with approximately 60° field of view, fastened to a common rigid bar with a disparity of about 5cm. The calibration target was a 380×304 mm commercial LCD monitor with a resolution of 1280×1024 pixels.

The cameras have been calibrated using a set of 20 shots and tested over a set composed of 40 different shots of the same active target. These shots have been acquired both with a single camera and with the complete camera pair, taking care to cover as much as possible of the respective fields of view.

Using the same data sets, we performed three different calibrations, using respectively the fully unconstrained model presented in Chapter 6 (Unconstrained), the non-

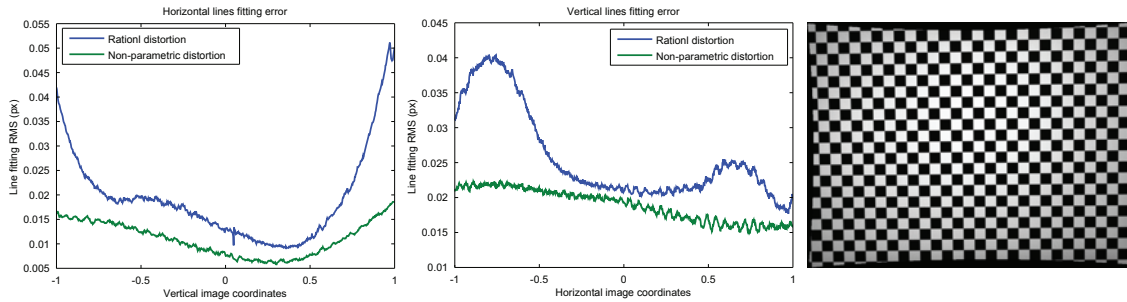


Figure 8.5: Left plot shows a comparison of our non-parametric distortion model compared with Fitzgibbon’s rational model. For a complete description of the displayed result see the text. On the right, an example of an undistorted image generated with our method.

parametric distortion proposed in this chapter (Non-Parametric) and the rational distortion (Rational).

Finally, we compared the performance of these three methods by means of two different experiments, assessing respectively the ability of providing a strictly projective virtual camera and to perform an accurate triangulation in the 3D space.

8.3.1 Image Undistortion

With this experiment we are testing the quality of the undistortion, that is how well the virtual pinhole camera obtained with the different methods approximate an ideal projective geometry. To this end, we exploited the projective invariance of straight lines. Specifically, for each horizontal and vertical scanline of the undistorted camera we collect the observed codes and we fit a straight line on them. Since we can assume the screen pixels to be regular and equally spaced, better pinhole approximation should exhibit a lower RMS error to the fitted line. Since a virtual pinhole cannot be produced with the fully unconstrained model, in Figure 8.5 we plotted only the results for the Non-Parametric and the Rational model. Each point in the plot is the average over 20 shots. While both methods are affected by some error, it is clear that the approximation given by the Non-Parametric approach yields less distorted lines. Furthermore, the structured nature of the RMS error obtained with the Rational model strongly suggests a systematic error due to the inability of the model to properly fit the data.

Note that, while our approach yields a much smaller undistortion error than the Rational model, there is still a strong spatial coherency in the error. This, composed with the fact that the structure of the error is coherent between the two models hits at the fact that the target itself might be not perfectly planar. The order of magnitude of this error is around 0.05mm, which is in line with current manufacturing standards.

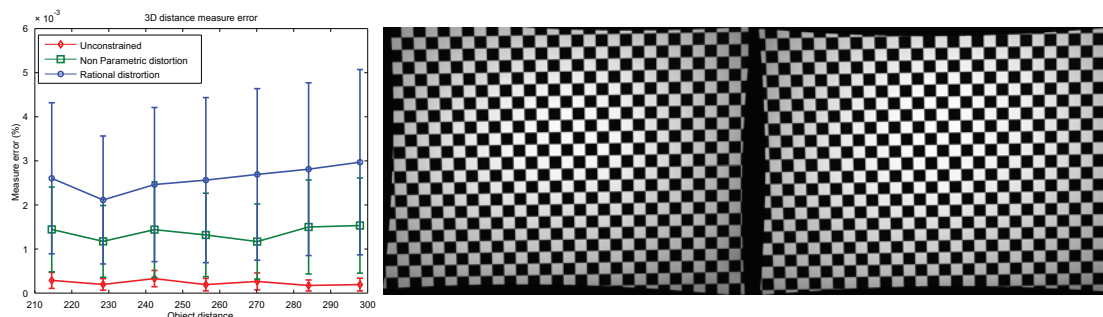


Figure 8.6: In the left-most plot we show the average relative error between the expected and measured distance of two triangulated points placed at a certain known distance. In the right we show an example of a stereo-rectified image pair.

8.3.2 3D Measurement

Our second experiment has been designed to investigate the quality of the calibration of the camera pair. After calibrating and rectifying the camera pair we used it to triangulate the 3D position of two random screen pixels observed by both cameras. We repeated the experiment for several shots with different positions of the target screen. In Figure 8.6 we plotted the average relative error between the expected and measured distance of two triangulated points with respect to the distance of the target from the camera pair. In this case, the Unconstrained model shows the best performance, in fact it produces a lower error at any distance and more repeatable measures. The Non-Parametric model exhibits a slightly higher error. This proves that the additional constraint hinders a perfect calibration. Still it is noticeably more reliable than the Rational model, thus it can be deemed as a reasonable alternative to the totally free model when high accuracy is needed, but it is not desirable to lose the advantages of the pinhole model.

Finally, in Figure 8.7 we give a qualitative example of a reconstructed range-map after stereo rectification provided by the OpenCV *stereoRectify* function and our calibration pipeline. The better alignment of the epipolar lines with the image rows gives a more precise and dense reconstruction.

8.4 Conclusions

In this chapter, we proposed a new calibration technique to model a central camera with a distortion described as a non-parametric dense displacement map on the input image. This approach combines the simplicity of a central camera model, enabling the usage of powerful projective geometry tools, while sharing the ability of unconstrained models to accommodate non-standard lens characteristics. Moreover, the independence of each ray entering the camera can be exploited to calibrate a stereo setup with



Figure 8.7: Reconstructed range-map triangulated from the OpenCV calibration and rectification (Left) and our proposed method (Right).

minor modifications on the process itself.

We assessed the performance of our method compared with current state-of-the-art parametric and completely unconstrained models, obtaining results lying in-between the two. Finally, we outperform the rational distortion model on mono camera calibration letting our technique be a viable choice for all applications for which a high accurate rectification of the camera is required.

III

Reconstruction and Measurement
Applications

9

Robust Multi-Camera 3D Ellipse Fitting

Ellipses are a widely used cue in many 2D and 3D object recognition pipelines. In fact, they exhibit a number of useful properties. First, they are naturally occurring in many man-made objects. Second, the projective invariance of the class of ellipses makes them detectable even without any knowledge of the acquisition parameters. Finally, they can be represented by a compact set of parameters that can be easily adopted within optimization tasks. While a large body of work exists in the literature about the localization of ellipses as 2D entities in images, less effort has been put in the direct localization of ellipses in 3D, exploiting images coming from a known camera network. In this chapter we propose a novel technique for fitting elliptical shapes in 3D space, by performing an initial 2D guess on each image followed by a multi-camera optimization refining a 3D ellipse simultaneously on all the calibrated views.

The proposed method is validated both with synthetic data and by measuring real objects captured by a specially crafted imaging head. Finally, to evaluate the feasibility of the approach within real-time industrial scenarios, we tested the performance of a GPU-based implementation of the algorithm.

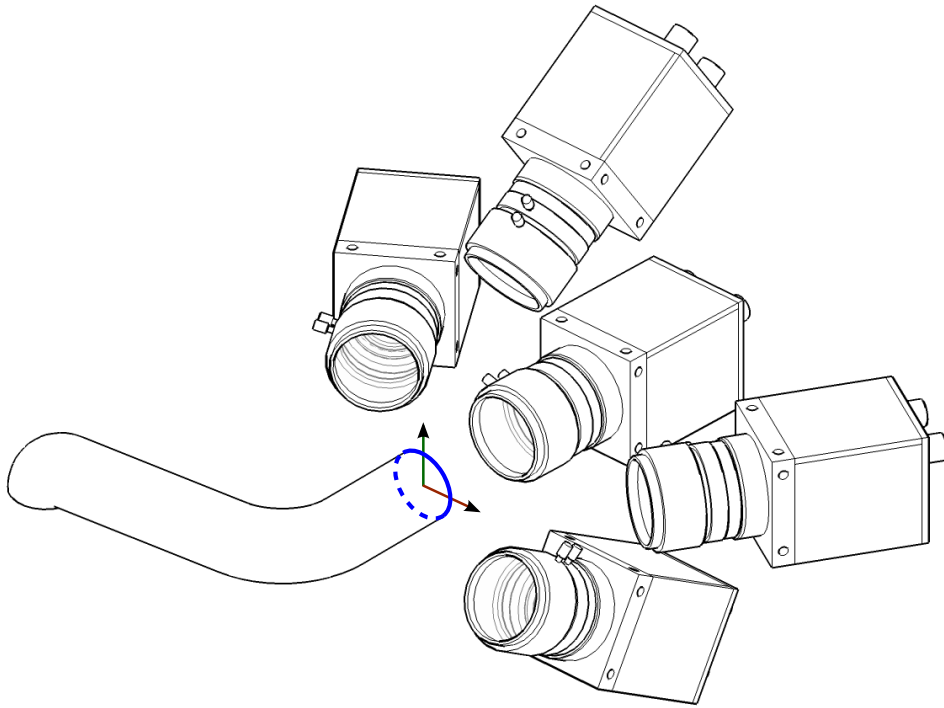


Figure 9.1: Schematic representation of a multi-camera system for industrial in-line pipes inspection.

9.1 Introduction

Among all the visual cues, ellipses offer several advantages that prompt their adoption within many machine vision tasks. To begin with, the class of ellipses is invariant to projective transformations, thus an elliptical shape remains so when it is captured from any viewpoint by a pinhole camera [59]. This property makes easy to recognize objects that contain ellipses [130, 82] or partially elliptical features [167]. When the parameters of one or more coplanar 3D ellipses that originated the projection are known, the class of homographies that make it orthonormal to the image plane can be retrieved. This is a useful step for many tasks, such as the recognition of fiducial markers [31, 140], orthonormalization of playfields [81], forensic analysis of organic stains [193] or any other planar metric rectification [52]. Furthermore, ellipses (including circles) are regular shapes that often appear in manufactured objects and can be used as optical landmarks for tracking and manipulation [200] or measured for accurate in-line quality assurance [160].

Because of their usefulness and broad range of applicability, it is not surprising that ellipse detection and fitting methods abound in the literature. In particular, when points belonging to the ellipse are known, they are often fitted through ellipse-specific least square methods [71]. In order to find co-elliptical points in images, traditional

parameter-space search schemas, such as RANSAC or Hough Transform, can be employed. Unfortunately, the significantly high dimensionality of 2D ellipse parametrization (which counts 5 degrees of freedom) makes the direct application of those techniques not feasible. For this reason a lot of efficient variants have appeared. Some try to reduce the number of samples for a successful RANSAC selection [166, 195]. Others attempt to escape from the curse of dimensionality that plagues the Hough accumulator [132, 53]. If high accuracy is sought, point-fitted ellipses can be used as an initial guess to be refined through intensity-based methods. Those approaches allow to obtain a sub-pixel estimation by exploiting the raw gradient of the image [147] or by preserving quantities such as intensity moments and gradients [87]. Multiple view geometry has also been exploited to get a better 3D ellipse estimation. In [182], multiple cameras are used to track an elliptical feature on a glove to obtain the estimation of the hand pose. The ellipses fitted in the images are triangulated with the algorithm proposed in [149] and the best pair is selected. In [128], holes in metal plates and industrial components are captured by a couple of calibrated cameras and the resulting conics are then used to reconstruct the hole in the Euclidean space. Also in [64] the intersection of two independently extracted conics is obtained through a closed form. All these approaches, however, exploit 3D constraints in an indirect manner, as triangulation always happens on the basis of the ellipses fitted over 2D data.

Here, we present a rather different technique that works directly in 3D space. Specifically, we adopt a parametric level-set approach, where the parameters of a single elliptical object that is observed by a calibrated network of multiple cameras (see Fig.9.1) are optimized with respect to an energy function that simultaneously accounts for each point of view. The goal of our method is to bind the 2D intensity and gradient-based energy maximization that happens within each image to a common 3D ellipse model. The performance of the solution has been assessed through both synthetic experiment and by applying it to a real world scenario. Finally, to make the approach feasible regardless of the high computational requirements, we propose a GPU implementation which performance has been compared with a well optimized CPU-based version.

9.2 Multiple Camera Ellipse Fitting

In our approach we are not seeking for independent optima over each image plane, as is the case with most ellipse fitting methods. Rather, our search domain is the parametrization of an ellipse in the 3D Euclidean space, and the optimum is sought with respect to its concurrent 2D reprojections over the captured images. In order to perform such optimization we need to sort out a number of issues. The first problem is the definition of a 3D ellipse parametrization that is well suitable for the task (that is, it makes easy to relate the parameters with the 2D projections). The second one, is the definition of an energy function that is robust and accounts for the usual cues for curve detection (namely the magnitude and direction of the intensity gradient). The last issue is the computation of the derivative of the energy function with respect to

the 3D ellipse parameters to be able to perform a gradient descent.

9.2.1 Parameterization of the 3D Ellipse

In its general case, any 2-dimensional ellipse in the image plane is defined by 5 parameters, namely: the length of the two axes, the angle of rotation and a translation vector with respect to the origin.

In matrix form it can be expressed by the locus of points $\vec{x} = (x_1 \ x_2 \ 1)^T$ in homogeneous coordinates for which the equation $\vec{x}^T A \vec{x} = 0$ holds, for

$$A = \begin{pmatrix} a & b & d \\ b & c & f \\ d & f & g \end{pmatrix} \quad (9.1)$$

with $\det(A) < 0$ and $ac - b^2 > 0$.

In the 3-dimensional case it is subjected to 3 more degrees of freedom (i.e. rotation around two more axes and the z-component of the translation vector). More directly, we can define the ellipse by first defining the plane T it resides on and then defining the 2D equation of the ellipse on a parametrization of such plane. In particular, let $\vec{c} = (c_1, c_2, c_3, 1)^T \in T$ be the origin of the parametrization, and $\vec{u} = (u_1, u_2, u_3, 0)^T$, $\vec{v} = (v_1, v_2, v_3, 0)^T$ be the generators of the linear subspace defining T , then each point on the 3D ellipse will be of the form $\vec{\delta} + \alpha \vec{u} + \beta \vec{v}$ with α and β satisfying the equation of an ellipse.

By setting the origin $\vec{\delta}$ to be at the center of the ellipse and selecting the directions \vec{u} and \vec{v} appropriately, we can transform the equation of the ellipse on the plane coordinates in such a way that it will take the form of the equation of a circle. Hence, allowing the 3D ellipse to be fully defined by the parametrization of the plane on which the ellipse resides. However, this representation has still one more parameter than the actual degrees of freedom of the ellipse. To solve this we can, without any loss of generality, set $u_3 = 0$, thus, by defining the matrix

$$\mathbf{U}_c = \begin{pmatrix} u_1 & v_1 & c_1 \\ u_2 & v_2 & c_2 \\ 0 & v_3 & c_3 \\ 0 & 0 & 1 \end{pmatrix} \quad (9.2)$$

and the vector $\vec{x} = (\alpha, \beta, 1)^T$, we can express any point p in the 3D ellipse as:

$$\vec{p} = \mathbf{U}_c \vec{x} \quad \text{subject to} \quad \vec{x}^T \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{pmatrix} \vec{x} = 0. \quad (9.3)$$

Even if \mathbf{U}_c embeds all the parameters needed to describe any 3d ellipse, it is often the case that an explicit representation through center \vec{c} and axes $\vec{a}_1, \vec{a}_2 \in R^3$ is needed. Let

\mathbf{U} be the 3×2 matrix composed by the first two columns of \mathbf{U}_c . The two axes \vec{a}_1, \vec{a}_2 can be extracted as the two columns of the matrix:

$$\mathbf{K} = \begin{pmatrix} | & | \\ \vec{a}_1 & \vec{a}_2 \\ | & | \end{pmatrix} = \mathbf{U}\phi^T$$

where ϕ^T is the matrix of left singular vectors of $\mathbf{U}^T\mathbf{U}$ computed via SVD decomposition. The vector \vec{c} is trivially composed by the parameters $(c_1 \ c_2 \ c_3)^T$.

Conversely, from two axes \vec{a}_1, \vec{a}_2 , the matrix \mathbf{U} can be expressed as:

$$\mathbf{U} = \mathbf{K} \begin{pmatrix} \alpha & -\beta \\ \beta & \alpha \end{pmatrix}$$

by imposing that $\begin{cases} \alpha K_{31} + \beta K_{32} = 0 \\ \alpha^2 + \beta^2 = 1 \end{cases}$. Finally, once \mathbf{U} has been computed, the 3D ellipse matrix can be composed in the following way:

$$\mathbf{U}_c = \begin{pmatrix} \mathbf{U} & \vec{c} \\ \vec{0} & 1 \end{pmatrix}$$

Finally, with this parametrization it is very easy to obtain the equation of the ellipse projected onto any camera. Given a projection matrix \mathbf{P} , the matrix \mathbf{A}_p describing the 2-dimensional ellipse after the projection can be expressed as:

$$\mathbf{A}_p = (\mathbf{P}\mathbf{U}_c)^{-T} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{pmatrix} (\mathbf{P}\mathbf{U}_c)^{-1} \quad (9.4)$$

9.2.2 Energy Function over the Image

To estimate the equation of the 3D-ellipse we set-up a level-set based optimization schema that updates the ellipse matrix \mathbf{U}_c by simultaneously taking into account its re-projection in every camera of the network. The advantages of this approach are essentially threefold. First, the equation of the 3D ellipse estimated and the re-projection in all cameras are always consistent. Second, erroneous calibrations that affects the camera network itself can be effectively attenuated, as shown in the experimental section. Third, the ellipse can be partially occluded in one or more camera images without heavily hindering the fitting accuracy.

In order to evolve the 3D ellipse geometry to fit the observation, we need to define the level set functions $\varphi_i : R^2 \rightarrow R$ describing the shape of the ellipse \mathbf{U}_c re-projected to the i^{th} camera. Given each level set, we cast the multiview fitting problem as the problem of maximizing the energy function:

$$E_{I_1 \dots I_n}(\mathbf{U}_c) = \sum_{i=1}^n E_{I_i}(\mathbf{U}_c) \quad (9.5)$$

Which sums the energy contributions of each camera:

$$E_{I_i}(\mathbf{U}_c) = \int_{R^2} \langle \nabla H(\varphi(\vec{x})), \nabla I_i(\vec{x}) \rangle^2 dx \quad (9.6)$$

$$= \int_{R^2} \langle H'(\varphi(\vec{x})) \nabla \varphi(\vec{x}), \nabla I_i(\vec{x}) \rangle^2 dx, \quad (9.7)$$

where H is a suitable relaxation of the Heavyside function. In our implementation, we used:

$$H(t) = \frac{1}{1 + e^{-\frac{t}{\sigma}}} \quad (9.8)$$

where parameter σ models the band size (in pixels) of the ellipse region to be considered. By varying σ we can manage the trade-off between the need of a regularization term in the energy function to handle noise in the image gradient and the estimation precision that has to be achieved.

The level set for a generic ellipse is rather complicated and cannot be easily expressed in closed form, however, since it appears only within the Heavyside function and its derivative, we only need to have a good analytic approximation in the boundary around the ellipse. We approximate the level set in the boundary region as:

$$\varphi_i(\vec{x}) \approx \frac{\vec{x}^T \mathbf{A}_i \vec{x}}{2\sqrt{\vec{x}^T \mathbf{A}_i^T \mathbf{I}_0 \mathbf{A}_i \vec{x}}} \quad (9.9)$$

Where $\mathbf{I}_0 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$ and \mathbf{A}_i is the re-projection of the ellipse \mathbf{U}_c into the i^{th} camera computed using equation (9.4). The function has negative values outside the boundaries of the ellipse, positive values inside and is exactly 0 for each point $\{\vec{x} | \vec{x}^T \mathbf{U}_c \vec{x} = 0\}$.

The gradient of the level set function $\nabla \varphi : R^2 \rightarrow R^2$ can actually be defined exactly in closed form:

$$\nabla \varphi_i(\vec{x}) = \frac{\mathbf{A}_i \vec{x}}{\sqrt{\vec{x}^T \mathbf{A}_i^T \mathbf{I}_0 \mathbf{A}_i \vec{x}}} \quad (9.10)$$

Starting from an initial estimation, given by a simple triangulation of 2d-ellipses between just two cameras, we maximize the energy function (9.5) over the plane parameters \mathbf{U}_c by means of a gradient scheme.

9.2.3 Gradient of the Energy Function

The gradient of the energy function can be computed as a summation of the gradient of each energy term. This gradient can be obtained by analytically computing the

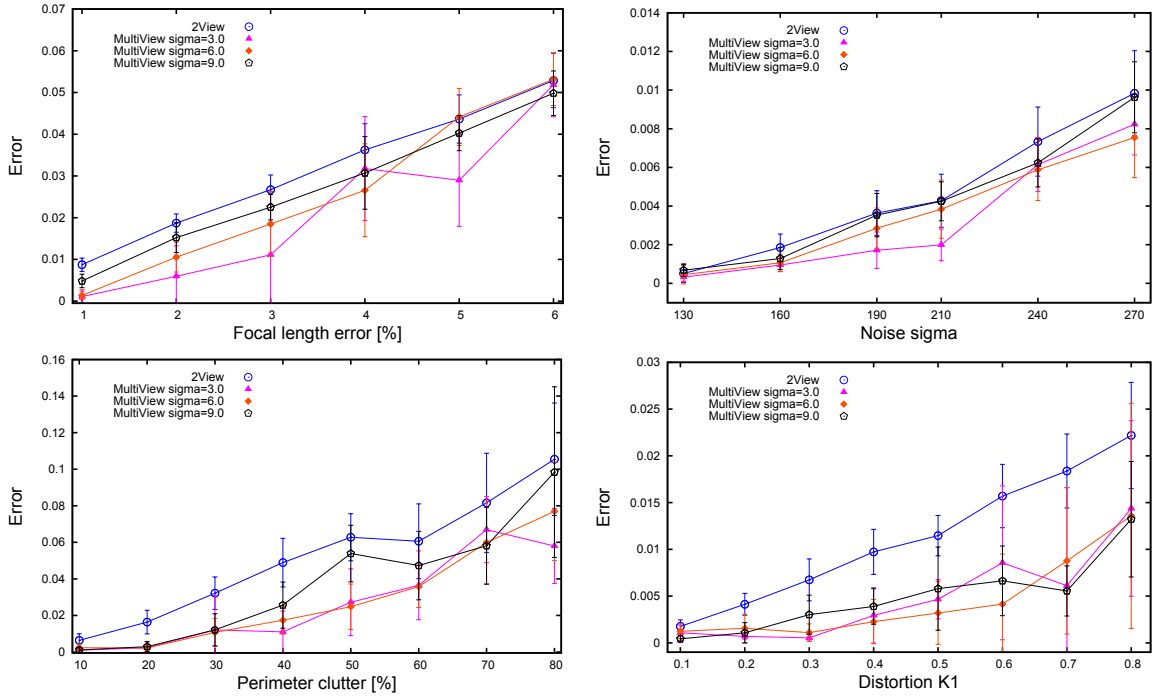


Figure 9.2: Evaluation of the accuracy of the proposed method with respect to different noise sources. The metric adopted is the relative error between the minor axis of the ground truth and of the fitted ellipse.

partial derivatives of equation (9.6) with respect to the eight parameters $(p_1 \dots p_8) = (u_1, v_1, c_1, u_2, v_2, c_2, v_3, c_3)$:

$$\begin{aligned} \frac{\partial}{\partial p_i} E_{I_i}(\mathbf{U}_c) &= \frac{\partial}{\partial p_i} \int_{R^2} E_{I_i}(\mathbf{U}_c, \vec{x})^2 dx \\ &= \int_{R^2} 2E_{I_i}(\mathbf{U}_c, \vec{x}) \frac{\partial}{\partial p_i} E_{I_i}(\mathbf{U}_c, \vec{x}) dx \end{aligned}$$

Where:

$$E_{I_i}(\mathbf{U}_c, \vec{x}) = \langle H'(\varphi(\vec{x})) \nabla \varphi(\vec{x}), \nabla I_i(\vec{x}) \rangle$$

and

$$\begin{aligned} \frac{\partial}{\partial p_i} E_{I_i}(\mathbf{U}_c, \vec{x}) &= \left(\frac{\partial}{\partial p_i} H'(\varphi(x)) \right) \langle \nabla \varphi(\vec{x}), \nabla I_i(\vec{x}) \rangle + \\ &\quad + H'(\varphi(\vec{x})) \left\langle \left(\frac{\partial}{\partial p_i} \nabla \varphi(\vec{x}) \right), \nabla I_i(\vec{x}) \right\rangle. \end{aligned}$$

The derivatives of the parametric level set functions can be computed analytically. At the beginning of each iteration we compute the derivative of the projected ellipse

matrices \mathbf{A}_i which are constant with respect to \vec{x} :

$$\frac{\partial}{\partial p_i} \mathbf{A}_i = \mathbf{T} + \mathbf{T}^T \quad (9.11)$$

where

$$T = \left(\frac{\partial}{\partial p_i} [(\mathbf{P}_i \mathbf{U}_c)^{-1}] \right)^T \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{pmatrix} (\mathbf{P}_i \mathbf{U}_c)^{-1} \quad (9.12)$$

and

$$\frac{\partial}{\partial p_i} [(\mathbf{P}_i \mathbf{U}_c)^{-1}] = -(\mathbf{P}_i \mathbf{U}_c)^{-1} (\mathbf{P}_i \frac{\partial}{\partial p_i} \mathbf{U}_c) (\mathbf{P}_i \mathbf{U}_c)^{-1}. \quad (9.13)$$

Then, using (9.11), we can compute the level set derivatives for each pixel:

$$\begin{aligned} \frac{\partial}{\partial p_i} \nabla \varphi(\vec{x}) &= \frac{(\frac{\partial}{\partial p_i} \mathbf{A}_i) \vec{x}}{\sqrt{\vec{x}^T \mathbf{A}_i^T \mathbf{I}_0 \mathbf{A}_i \vec{x}}} - \\ &\quad - \frac{\mathbf{A}_i \vec{x} (\vec{x}^T (\frac{\partial}{\partial p_i} \mathbf{A}_i)^T \mathbf{I}_0 \mathbf{A}_i \vec{x} + \vec{x}^T \mathbf{A}_i^T \mathbf{I}_0 (\frac{\partial}{\partial p_i} \mathbf{A}_i) \vec{x})}{2(\vec{x}^T \mathbf{A}_i^T \mathbf{I}_0 \mathbf{A}_i \vec{x})^{\frac{3}{2}}} \end{aligned} \quad (9.14)$$

$$\frac{\partial}{\partial p_i} \varphi(\vec{x}) = \frac{1}{2} \langle \vec{x}, \frac{\partial}{\partial p_i} \nabla \varphi(\vec{x}) \rangle \quad (9.15)$$

$$\frac{\partial}{\partial p_i} H'(\varphi(\vec{x})) = H''(\varphi(\vec{x})) \frac{\partial}{\partial p_i} \varphi(\vec{x}). \quad (9.16)$$

By summing the derivative $\frac{\partial}{\partial p_i} E_{I_i}(\mathbf{U}_c, \vec{x})$ over all images and all pixels in the active band in each image, we obtain the gradient $\mathbf{G} = \nabla E_{I_1 \dots I_n}(\mathbf{U}_c)$. At this point, we update the 3D ellipse matrix \mathbf{U}_c through the gradient step

$$\mathbf{U}_c^{(t+1)} = \mathbf{U}_c^{(t)} + \eta \mathbf{G} \quad (9.17)$$

where η is a constant step size.

9.3 Experimental evaluation

We evaluated the proposed approach both on a set of synthetic tests and on a real world quality control task where we measure the diameter of a pipe with a calibrated multi-camera setup. In both cases, lacking a similar 3D based optimization framework, we compared the accuracy of our method with respect to the results obtained by triangulating ellipses optimally fitted over the single images.

The rationale of the synthetic experiments is to be able to evaluate the accuracy of the measure with an exactly known ground truth (which is very difficult to obtain on real objects with very high accuracy). Further, the synthetically generated imagery



Figure 9.3: The experimental Multiple-camera imaging head.

permits us to control the exact nature and amount of noise, allowing for a separate and independent evaluation for each noise source.

By contrast, the setup employing real cameras does not give an accurate control over the scene, nevertheless it is fundamental to assess the ability of the approach to deal with the complex set of distractors that arise from the imaging process (such as reflections, variable contrast, defects of the object, bad focusing and so on).

In both cases the ellipse detection is performed by extracting horizontal and vertical image gradients with an oriented derivative of Gaussian filter. Edge pixels are then found by non-maxima suppression and by applying a very permissive threshold (no hysteresis is applied). The obtained edge pixels are thus grouped into contiguous curves, which are in turn fitted to find ellipses candidates. The candidate that exhibits the higher energy is selected and refined using [147]. The refined ellipses are then triangulated using the two images that score the lower triangulation error. The obtained 3D ellipse is finally used both as the result of the baseline method (labeled as *2view* in the following experiments) and as the initialization ellipse for our refinement process (labeled as *multiview*).

All the experiments have been performed with 3Mp images and the processing is done with a modern 3.2 Ghz Intel Core i7 PC equipped with Windows 7 Operating System. The CPU implementation was written in C++ and the GPU implementation uses the CUDA library. The video card used was based on the Nvidia 670 chipset with 1344 CUDA cores.

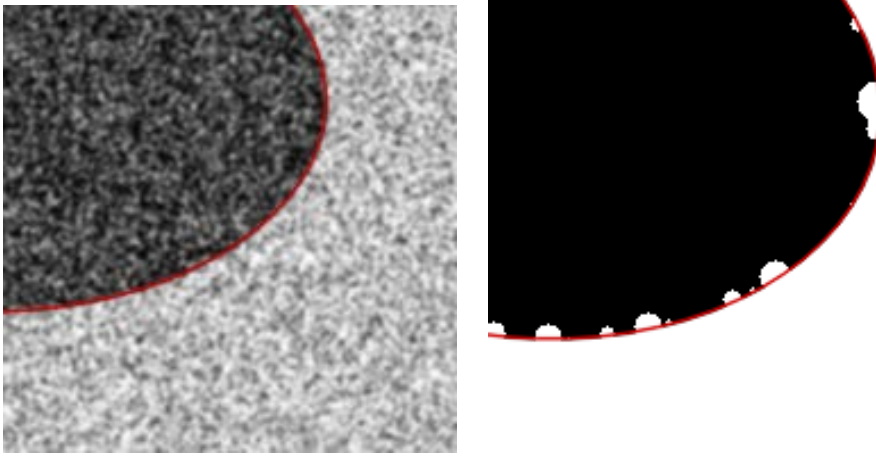


Figure 9.4: Examples of images with artificial noise added. Respectively additive Gaussian noise and blur in the left image and occlusion in the right image. The red line shows the fitted ellipse.

9.3.1 Synthetic Experiments

For this set of experiments we chose to evaluate the effect of four different noise sources over the optimization process. Specifically, we investigated the sensitivity of the approach to errors on the estimation of the focal length and of the radial distortion parameters of the camera and the influence of Gaussian noise and clutter corrupting the images. In Fig. 9.4 examples of Gaussian noise and clutter are shown (note that these are details of the images, in the experiments the ellipse was viewed in full).

For each test we created 5 synthetic snapshots of a black disc as seen from 5 different cameras looking at the disk from different points of view (see Fig. 9.1 and Fig. 9.3). The corruption by Gaussian noise has been produced by adding to each pixel a normal distributed additive error of variable value of σ , followed by a blurring of the image with a Gaussian kernel with $\sigma = 6$. The artificial clutter has been created by occluding the perimeter of the disc with a set of random white circles until a given percentage of the original border was corrupted. This simulates the effect of local imaging effect such as the presence of specular highlights that severely affect the edge detection process. The focal length error was obtained by changing the correct focal length of the central camera by a given percentage. Finally, the distortion error was introduced by adding an increasing amount to the correct radial distortion parameter $K1$.

In Fig. 9.2 we show the results obtained using the baseline triangulation and our optimization with different values of the parameter σ used for the heavyside function (respectively 3, 6 and 9 pixels). As expected, in all the tests performed the relative error grows with the level of noise. In general, all the methods seem to be minimally sensitive to Gaussian noise, whereas the clutter has a big effect even at low percentages. The baseline method performs consistently worse and, among the multiview configurations, the one with lower heavyside band appears to be the most robust for almost

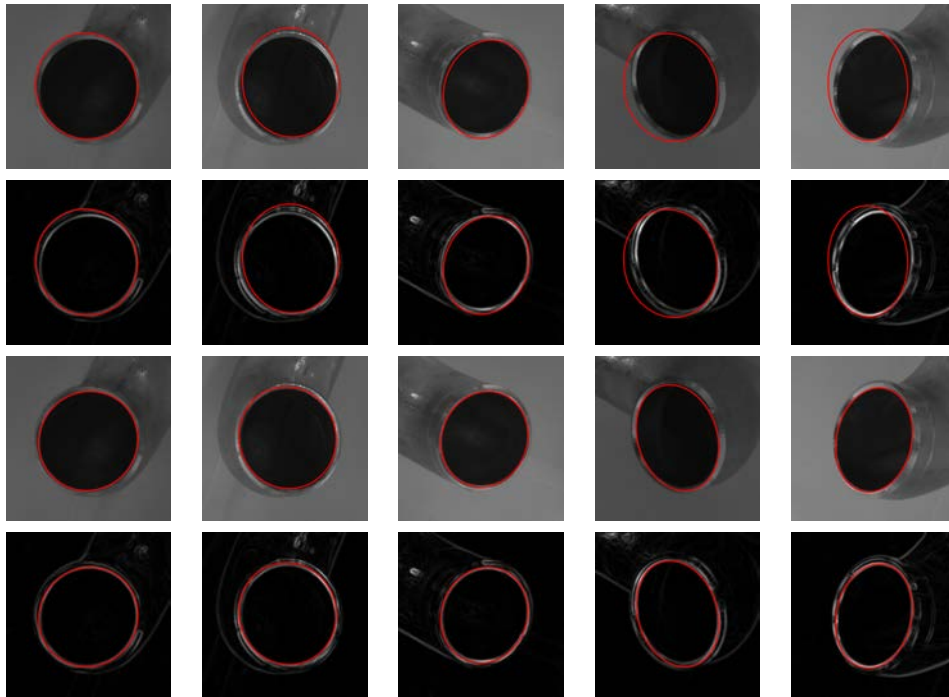


Figure 9.5: Comparison between the accuracy of the initial 2D fitting and the proposed 3D optimization.

all noise levels. This is probably due to the fact that the images have already been smoothed by the gradient calculation step, and thus further smoothing is not required and, to some degree, leads to a more prominent signal displacement.

9.3.2 Real World Application

For the experiments with real images we built an imaging device that hold 5 PointGrey Flea3 3.2Mp Monochrome USB3 machine vision cameras (see Fig. 9.3). The 5 cameras were calibrated for both intrinsic and extrinsic parameters. We used an aluminum pipe for air conditioning system as the object to be measured, and the imaging head has been supplemented with four high power leds in order to get an even illumination of the rim. This is a typical scenario for in-line inspection in manufacturing lines. Additionally, the smooth and polished surface of the pipe offers especially challenging conditions for ellipse detection and refinement, since reflections and changes of contrast tend to create a lot of false elliptical sectors and some highly structured noise.

If Fig. 9.5 a complete qualitative example of the refinement process is shown. In the first two rows of the figure the reprojection of the initially estimated 3D ellipse is overlaid to both the original images and the intensity-coded gradient magnitude. In the remaining rows the reprojection of the optimized 3D ellipse is overlaid over the same images. The images used for the initial triangulation in this specific case were the first

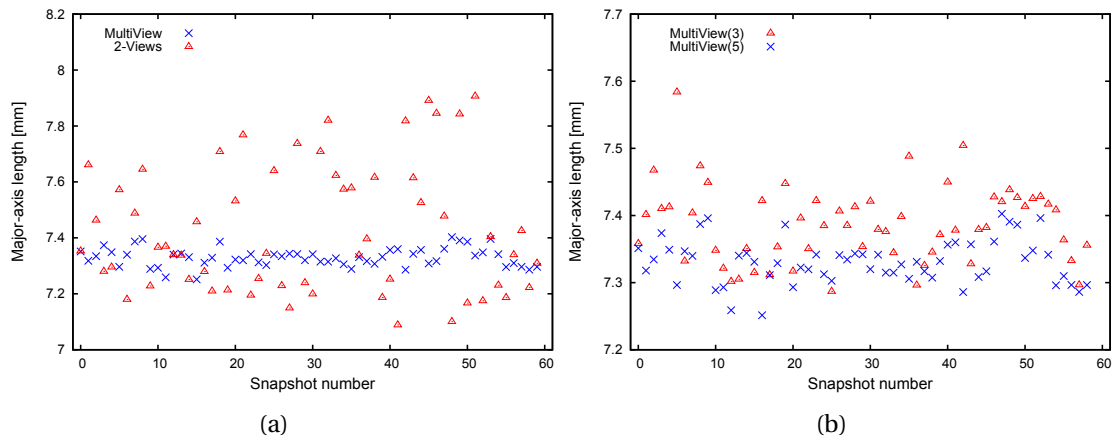


Figure 9.6: Quantitative assessment of the improvement in accuracy (Left) and effect of the number of views over the measure quality.

and the third. Specifically, the initial guess for the first image was a slightly off-center ellipse fitted in between the edge response produced by the inner and outer rims of the pipe opening (see the gradient image). As a matter of fact, it is immediate to note that these two images exhibits the lower reprojection error, especially for the central camera. However, the other reprojections are rather grossly misaligned with the remaining three points of view. By contrast, almost all the misalignment has been corrected after performing the 3D refinement procedure. While some degree of displacement is still visible in some images, we think that this is mainly due to miscalibration of the extrinsic parameters of the imaging head.

We manually measured the internal and external diameter of the pipe with a caliper (with ± 0.1 mm accuracy) obtaining respectively 13.9 and 16.1 mm. However, since the optimization process aim to converge toward the middle of the two rims, it would make no sense to evaluate directly the measurement error committed. Still, the standard deviation of the data with respect to several subsequent measures of the same object from slightly different angles can be considered a good indication of measurement error. Indeed, even if the final measure can be affected by systematic errors, they can be estimated and corrected a-posteriori. In Fig. 9.6(a) we plotted the measured length of the major axis of the detected 3D ellipse for 60 repeated shots of the pipe opening. The improvement in uncertainty reduction after the refinement step is clearly noticeable as the variance of the measurements is strongly reduced. Indeed, the standard deviation went from 0.23 to 0.03.

All the refinements performed so far have been conducted using 5 points of view. In order to complete our tests it would have been interesting to evaluate if similar accuracy could be obtained using a smaller number of cameras. To this end we disabled two cameras and took further 60 shots of the pipe. The results are plotted in Fig. 9.6(b). While the dispersion of the measurements is a little higher using only three points of

view, it is still noticeably smaller than the one obtained without the optimization step (note that the scales of Fig. 9.6(a) and Fig. 9.6(b) are different).

9.4 GPU-based Implementation

In a naive implementation, the optimization scheme proposed is quite intensive in terms of raw computing power. Especially for the gradient computation, which requires several matrix and vector multiplications that may easily sum up to an unacceptable total computation time.

However, the intrinsic structure of the problem leads naturally to an implementation in which every pixel in the area of interest defined by the Heaviside function are computed in parallel. After this computation, that can be performed with no required synchronization between each view, a reduction step is needed to aggregate all terms and obtain the final value of the energy and gradient in each iteration.

We implemented the algorithm in C++ with no additional external libraries except for OpenCV for image IO, OpenMP for CPU parallelization and CUDA for GPU computing. Both the CPU and GPU based implementations are essentially the same, except for the fact that the latter can exploit the massive computing power of modern graphics cards. For every algorithm's iteration, a CPU-based function computes a list of pixel for each image that will be affected by the computation. This list is generated by considering a band around each 2d-ellipse reprojection with a thickness of 5σ pixels and is uploaded to the device memory, together with the optimized parameters and the pre-computed image gradient for each pixel in the list. Once the upload is completed, all available stream processors are used to compute the energy and the energy gradient terms. At the end of the computation steps, all threads are synchronized and 9 values are reduced (energy and the 8 terms of the gradient) to obtain the final values. The total energy is used to track the optimization status and trigger a termination criteria, the gradient is used to adjust the 3d ellipse that is being optimized, moving toward a local maxima.

We tested the execution time per iteration for both the CPU and GPU based implementation of our algorithm (see Fig.9.7) with respect to the average number of pixel processed. In both cases, the process is fast enough to handle a real-time optimization in 3 megapixels images with the fitted ellipse spanning into about 50% of the image.

As expected, the GPU implementation performs better than the CPU and exhibits a more consistent running time throughout the tests. This is probably due to the fact that we are dealing with a dedicated hardware. Finally, the synchronization overhead caused by the reductions decreases the performance gap between the two implementations when a relatively low number of pixels are processed, which in turn becomes dramatic when an optimization involving more than 10^5 pixels is needed.

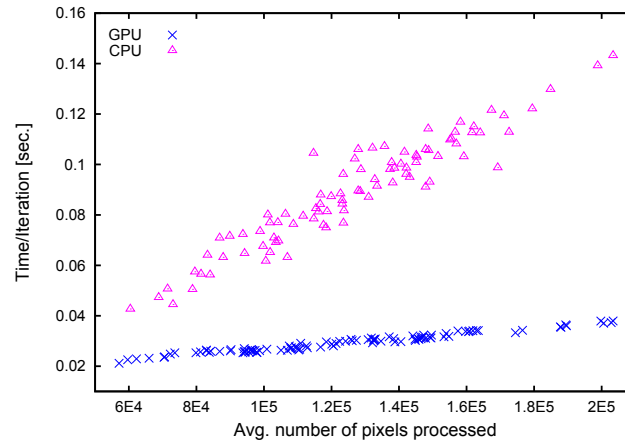


Figure 9.7: Comparison between the running time of the CPU and GPU-based implementations of the multiview algorithm. Times are plotted with respect to the number of pixels in the evaluated mask (i.e. size of the ellipse to be refined).

9.5 Conclusions

In this chapter we presented a novel approach for ellipse fitting that exploits multiple simultaneous calibrated views of the same physical elliptical object. The proposed technique starts by obtaining an initial estimation of the 3D ellipse using 2D based fitting methods followed by a pairwise triangulation. This initial guess is then refined by moving its parameters according to an intensity-based level set approach that accounts for all the images simultaneously. To this end, a specially crafted parametrization and an apt energy function have been introduced. The effectiveness of the approach has been evaluated through an extensive experimental section that shows its resilience to a wide range of noise sources and compares the accuracy obtained with respect to a baseline approach. Finally, a running time analysis of our GPU-based implementation of the algorithm shows that the proposed method can be effectively used in real-time applications.

10

Simultaneous Optical Flow and Dichromatic Parameter Recovery

In this chapter we propose an approach for the recovery of the dichromatic model from two views, i.e., the joint estimation of illuminant, reflectance, and shading of each pixel, as well as the optical flow between the images. The approach is based on the minimization of an energy functional linking the dichromatic model to the image appearances and the flow between the images to the factorized reflectance component. In order to minimize the resulting under-constrained problem, we apply vectorial total variation regularizers both to the common scene reflectance, and to the flow hyperparameters, enforcing the physical priors that the reflectance is constant in all images across a uniform material, and that the flow varies smoothly within the same rigid object. We show the effectiveness of the approach compared with single view model recovery both in terms of model constancy and of closeness to the ground truth.

10.1 Introduction

We depart from the dichromatic model (See Section 2.5) so as to describe the image radiance as a combination of shading, specular highlights, surface reflectance and the illuminant power spectrum. Our multi-view dichromatic parameter recovery method separates the scene illuminant, shading and object surface reflectance by minimising the total variation subject to the notion that the object reflectance and illuminant power spectrum across the image sequence under consideration should not change. This also imposes further constraints on the optical flow which are akin to those imposed upon brightness in trichromatic imagery [29]. This leads to an optimisation problem where a regularisation approach is used to enforce the consistency of the scene photometric parameters over an image sequence. This contrasts with previous approaches where the intersection of dichromatic planes [69, 177], assumed chromaticities of common light sources [69], or structural optimisation [97] are used on single images.

The contribution of this work is twofold: First, to the best of our knowledge, this is the first approach that uses reflectance constancy across multiple images to improve the recovery of the dichromatic parameters, relating the reflectance to the optical flow between multiple images. Second, we introduce a novel affine hyper-prior for the flow similar in spirit to the one presented in [114], but that in combination with a Total Variation regularization provides a natural piecewise-rigid assumption on the motion, resulting in an improvement of the optical flow estimation in parallel with the improvement to the photometric parameters. Experiments show that the approach is capable of providing a more stable recovery of illuminant, reflectance, shading and specular parameters with respect to the state of the art.

10.2 Multi-view Dichromatic Parameter Recovery

In this section we will present the multi-view dichromatic model energy functional joining the dichromatic factorization on each image with the coherency of the reflectance across corresponding points in multiple images.

10.2.1 Multi-Spectral Imaging and the Dichromatic Model

As mentioned earlier we employ the dichromatic model [162]. By assuming a uniform illuminant power spectrum across the scene, the dichromatic model expresses the image radiance $I(\mathbf{u}, \lambda)$ at pixel location $\mathbf{u} = (u_1, u_2)$ and wavelength λ as follows:

$$I(\mathbf{u}, \lambda) = g(\mathbf{u})L(\lambda)S(\mathbf{u}, \lambda) + k(\mathbf{u})L(\lambda) \quad (10.1)$$

where $L(\lambda)$ and $S(\mathbf{u}, \lambda)$ are the illuminant power spectrum and surface reflectance at wavelength λ , respectively, $g(\mathbf{u})$ is the shading factor governing the proportion of diffuse light reflected from the object and $k(\mathbf{u})$ is the specular coefficient at pixel \mathbf{u} .

Here, we also make the assumption that $\sum_{\lambda} S(\mathbf{u}, \lambda)^2 = 1$. Note that this can be done without any loss of generality since the illuminant power spectrum can be normalized such that the shading factor and specular coefficients are rescaled accordingly.

10.2.2 Optical Flow and Reflectance Coherency

One of the main features of the dichromatic model is that the reflectance $S(\mathbf{u}, \lambda)$ is a characteristic of the object's material, being invariant to the geometry of the object and its relative position with respect to the light source and the viewer. As a consequence, it is preserved across multiple images. We model this correspondence in a two image setting by maintaining one single reflectance function on one image and relating it to the reflectance on a second image through an optical flow function $f(\mathbf{u}) = \mathbf{u}' : \Omega_1 \rightarrow \Omega_2$ which maps points from the first to the second image.

Note that, for the computation of the flow, it is often assumed that the image brightness remains approximately unchanged across the two views under consideration. The “constant” brightness assumption applies to stereo and multiple-view settings where the baseline is not overly wide. While effective, this can introduce errors in the estimation of the reflectance about specular spikes and lobes in the scene or where there is a big change in the relative angle between the objects in the scene and the illuminant direction. Further, for highly specular pixels, the reflectance information is effectively lost at capture, *i.e.* $g(\mathbf{u})L(\lambda)S(\mathbf{u}, \lambda) \approx 0$. For this reason, in our formulation, we make use of the multiplicative gating function

$$W(\mathbf{u}) = \exp(-\tau \|I(\mathbf{u}, \lambda) - \mathcal{P}(I(\mathbf{u}, \lambda))\|) \quad (10.2)$$

where $\mathcal{P}(I(\mathbf{u}, \lambda))$ is the projection of the image radiance $I(\mathbf{u}, \lambda)$ onto the dichromatic plane [68] spanned by the radiance over the neighbourhood about pixel location \mathbf{u} . The dichromatic plane can be computed using SVD [153].

The gating function above reflects the observation that, as the deviation of the image radiance from the dichromatic plane increases, the diffuse reflection decreases in importance [68]. Therefore, the function $W(\mathbf{u})$ can be viewed as a weight in the illuminant and reflectance recovery error when applied to the dichromatic energy terms:

$$E_{DI_1} = \int_{\Omega_1} W_1(\mathbf{u})^2 \sum_{\lambda} \left(I_1(\mathbf{u}, \lambda) - L(\lambda) \left(g_1(\mathbf{u})S(\mathbf{u}, \lambda) + k_1(\mathbf{u}) \right) \right)^2 d\mathbf{u} \quad (10.3)$$

$$E_{DI_2} = \int_{\Omega_1} W_2(\mathbf{u}')^2 \sum_{\lambda} \left(I_2(\mathbf{u}', \lambda) - L(\lambda) \left(g_2(\mathbf{u}')S(\mathbf{u}, \lambda) + k_2(\mathbf{u}') \right) \right)^2 d\mathbf{u} \quad (10.4)$$

where the subscript indicate the index for either of the two images. Note that even for the term related to the second image the integration is performed over the domain Ω_1 of the first image whereby the relations with Ω_2 is always mediated through the flow f .

Moreover, the gating function $W(\mathbf{u})$ decreases in value for increasingly specular pixels. This is in accordance with the dichromatic plane formalism used to define $W(\mathbf{u})$, which implies that, for specular highlights, the gating function tends to zero,

i.e. the gating function and the specular coefficient are nearly orthogonal with respect to each other. Hence, in equations (10.3) and (10.4), the contribution of the specular pixels to the energy functional is negligible. As a result, we remove the specular coefficient from further consideration for purposes of our optimization approach and, instead, compute it analytically at the end of the process, once the reflectance, illuminant power spectrum, and shading are in hand.

10.2.3 Total Variation Regularization

Our goal is, to minimize the energy terms over the flow $f(\cdot)$ and the dichromatic model parameters. However, the problem is under determined, problem that we address adding regularization terms to the energy functional above. The *Total Variation* (TV) of a function $f : \mathbb{R}^m \supseteq \Omega \rightarrow \mathbb{R}^n$ is an operator defined as

$$\text{TV}(f) = \sup_{p_1, \dots, p_m} \left\{ \int_{\Omega} \sum_{i=1}^n f_i(\mathbf{x}) \nabla \cdot p_i(\mathbf{x}) \, d\mathbf{x} : p_1, \dots, p_m \in \mathcal{C}^1(\Omega, \mathbb{R}^n) \right\}, \quad (10.5)$$

where $\mathcal{C}^1(\Omega, \mathbb{R}^n)$ is the set of continuously differentiable functions from Ω to \mathbb{R}^n , and p_1, \dots, p_m satisfy $\sum_{i=1}^m \|p_i(x)\|^2 \leq 1$ everywhere except at most in a subset of measure 0. Further, if f is a differentiable function, the TV assumes the equivalent form

$$\text{TV}(f) = \int_{\Omega} \|Df(x)\|_2 \, dx, \quad (10.6)$$

where Df is the differential or Jacobian matrix of f and $\|\cdot\|_2$ denotes the Frobenius norm.

Used as a regularizer, TV privileges piecewise constant solutions and for this property has found a multitude applications ranging from image processing restoration [155], to segmentation [148], to the estimation of the optical flow [192, 63]. Here we adopt TV to impose smoothness priors on the reflectance and flow estimates. The reflectance component is assumed to be constant over image patches of uniform material, thus TV is directly applicable to S , seen as a function from Ω_1 to \mathbb{R}^{ℓ} where ℓ is the number of spectral bands.

For the flow, however, there is no reason to assume a piecewise constant model. Most approaches in the literature opt to express the flow as a displacement $f(\mathbf{u}) = \mathbf{u} + T(\mathbf{u})$ where the displacement is regularized, resulting in a piecewise uniform translation. Here we opt for a higher order smoothness prior, where the displacement is assumed to be locally affine

$$f(\mathbf{u}) = \mathbf{u} + A(\mathbf{u})\mathbf{u} = \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} + \begin{pmatrix} a_1(\mathbf{u}) & a_2(\mathbf{u}) & a_3(\mathbf{u}) \\ a_4(\mathbf{u}) & a_5(\mathbf{u}) & a_6(\mathbf{u}) \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} u \\ v \\ 1 \end{pmatrix}. \quad (10.7)$$

This local affine model can be seen as an approximation of view transformation under a weak camera model of local planar patches, and thus can be assumed to be piecewise uniform on much larger patches within a rigidly moving object. As a result the

hyperparameters $a_1(\mathbf{u}), \dots, a_6(\mathbf{u})$ can be assumed to be piecewise constant within such patches.

Finally, we perform a convex relaxation of the total variation functional [39] transforming the TV regularized optimization problem

$$\min_f E(f) + TV(f) \quad (10.8)$$

into the relaxed problem

$$\min_{f, f_{TV}} E(f) + \int \frac{\|f - f_{TV}\|^2}{\delta} + TV(f_{TV}). \quad (10.9)$$

While the size increases with the addition of the auxiliary function f_{TV} , assuming $E(f)$ convex, the formulation becomes convex for $\delta > 0$ and converges to the original variational problem as $\delta \rightarrow 0$.

10.2.4 Multi-view dichromatic functional

Assembling the data fidelity terms and the regularizers, we obtain the energy Multi-view dichromatic functional

$$E = \alpha (\mu E_{DI_1} + (1 - \mu) E_{DI_2}) \quad (10.10)$$

$$+ \rho_S \left(\int_{\Omega_1} \frac{\|S(\mathbf{u}) - S_{TV}(\mathbf{u})\|^2}{\delta_S} d\mathbf{u} + \int_{\Omega_1} \|DS_{TV}(\mathbf{u})\|_2 d\mathbf{u} \right) \quad (10.11)$$

$$+ \rho_f \left(\int_{\Omega_1} \frac{\|A(\mathbf{u}) - A_{TV}(\mathbf{u})\|_2^2}{\delta_f} d\mathbf{u} + \int_{\Omega_1} \|DA_{TV}(\mathbf{u})\|_2 d\mathbf{u} \right) \quad (10.12)$$

which is then minimized over $S, f, L, g_1, g_2, S_{TV}$, and f_{TV} to obtain simultaneous flow estimation and joint factorization of the dichromatic model over the two images. Here α, ρ_S , and ρ_f are constants balancing the data fidelity and regularization terms, while $\mu \in [0; 1]$ is used to limit the effect that errors in the estimation of the flow can have in the dichromatic factorization originating from the second image. Note that, as mentioned earlier, due to the $W(\mathbf{u})k(\mathbf{u})$ orthogonality we can eliminate the $W(\mathbf{u})k(\mathbf{u})$ terms thus avoiding the minimization over k , and recover the specular coefficient after the optimization from the optimal illuminant, reflectance, and shading with the relation $k(\mathbf{u}) = \frac{1}{\ell} \sum_{\lambda} \frac{I(\mathbf{u}, \lambda)}{L(\lambda)} - g(\mathbf{u})S(\mathbf{u}, \lambda)$.

10.3 Minimization Process

To optimize E we adopt an alternating minimization procedure, rotating through the following steps:

1. Minimize with respect to $L(\lambda)$, $g_1(\mathbf{u})$, and $g_2(f(\mathbf{u}))$, keeping $S(\mathbf{u}, \lambda)$, $f(\mathbf{u})$, $S_{TV}(\mathbf{u}, \lambda)$ and $A_{TV}(\mathbf{u})$ fixed;
2. Update $S(\mathbf{u}, \lambda)$ and $f(\mathbf{u})$ through a gradient descent step, keeping all other variables fixed;
3. Minimize (10.11) and (10.12) to obtain a new estimate of $A_{TV}(\mathbf{u})$ and $S_{TV}(\mathbf{u})$.

For the first step, differentiating E with respect to $g_1(\mathbf{u})$ and $g_2(f(\mathbf{u}))$, and setting both equations equal to zero we obtain

$$g_1(\mathbf{u}) = \frac{\sum_{\lambda} I_1(\mathbf{u}, \lambda) S(\mathbf{u}, \lambda) L(\lambda)}{\sum_{\lambda} S(\mathbf{u}, \lambda)^2 L(\lambda)^2} \quad (10.13)$$

$$g_2(f(\mathbf{u})) = \frac{\sum_{\lambda} I_2(f(\mathbf{u}), \lambda) S(\mathbf{u}, \lambda) L(\lambda)}{\sum_{\lambda} S(\mathbf{u}, \lambda)^2 L(\lambda)^2}. \quad (10.14)$$

Differentiating $E = \mu E_{DI_1} + (1 - \mu) E_{DI_2} + \text{const.}$ with respect to $L(\lambda)$ and setting equal to zero, we have:

$$L(\lambda) = C_1 \frac{\int_{\Omega_1} S(\mathbf{u}, \lambda) \Delta_I(\mathbf{u}, \lambda) d\mathbf{u}}{\int_{\Omega_1} S(\mathbf{u}, \lambda)^2 \Delta_S(\mathbf{u}, \lambda) d\mathbf{u}}, \quad (10.15)$$

where

$$\begin{aligned} \Delta_I(\mathbf{u}, \lambda) &= \mu W_1(\mathbf{u})^2 I_1(\mathbf{u}, \lambda) g_1(\mathbf{u}) + (1 - \mu) W_2(f(\mathbf{u}))^2 I_2(f(\mathbf{u}), \lambda) g_2(f(\mathbf{u})) \\ \Delta_S(\mathbf{u}, \lambda) &= \mu W_1(\mathbf{u})^2 g_1(\mathbf{u})^2 + (1 - \mu) W_2(f(\mathbf{u}))^2 g_2(f(\mathbf{u}))^2 \end{aligned}$$

and C_1 is a normalizing constant satisfying $\sum_{\lambda} L(\lambda)^2 = 1$.

Hence, for the first step of the optimization process, we find the global optimum of E with respect to $g_1(\mathbf{u})$, $g_2(f(\mathbf{u}))$, and $L(\lambda)$ by alternating Equations (10.13), (10.14), and (10.15). Note that, while we are estimating $g_1(\mathbf{u})$ in the regular lattice of the first image, we are estimating $g_2(f(\mathbf{u}))$ through f . This means that, in a discrete image setting, the estimated values of the second image's shading factor are not aligned with that image's regular lattice, but are shifted according to the flow f . This is not in general a problem because only those point are used in the energy computation, but, as we will see, an interpolation step will be needed in the update of f .

For the second step, we compute the gradient of E with respect to the reflectance $S(\mathbf{u}, \lambda)$ and the hyper-parameter $A(\mathbf{u})$. Note that the the data fidelity term only depends of $f(\mathbf{u})$, thus, using the chain rule for the data fidelity term only, we can write

$$\partial_{A(\mathbf{u})} E = (\partial_{f(\mathbf{u})} E) (\partial_{A(\mathbf{u})} f(\mathbf{u})) + \rho_f \frac{A(\mathbf{u}) - A_{TV}(\mathbf{u})}{\delta_f}. \quad (10.16)$$

Here

$$\partial_{A(\mathbf{u})} f(\mathbf{u}) = \partial_{(a_1(\mathbf{u}), \dots, a_6(\mathbf{u}))} f(\mathbf{u}) = \begin{pmatrix} u_1 & u_2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & u_1 & u_2 & 1 \end{pmatrix} \quad (10.17)$$

while

$$\begin{aligned} \partial_{f(\mathbf{u})}E &= \alpha(1-\mu) \left[W_2(\mathbf{u}') \sum_{\lambda} 2 \left(I_2(\mathbf{u}', \lambda) - L(\lambda) g_2(\mathbf{u}') S(\mathbf{u}, \lambda) \right) \right. \\ &\quad \left. \cdot \left(\partial_{f(\mathbf{u})} I_2(\mathbf{u}', \lambda) - L(\lambda) S(\mathbf{u}, \lambda) \partial_{f(\mathbf{u})} g_2(\mathbf{u}') \right) + E_{\text{DL}_2}(\mathbf{u}) \partial_{f(\mathbf{u})} W_2(\mathbf{u}') \right]. \end{aligned} \quad (10.18)$$

Furthermore, the energy gradient with respect to reflectance can be expressed as:

$$\begin{aligned} \partial_{S(\mathbf{u}, \lambda)}E &= -2\alpha\mu g_1(\mathbf{u}) W_1(\mathbf{u}) L(\lambda) \left(I_1(\mathbf{u}, \lambda) - g_1(\mathbf{u}) L(\lambda) S(\mathbf{u}, \lambda) \right) \\ &\quad -2\alpha(1-\mu) g_2(\mathbf{u}') W_2(\mathbf{u}') L(\lambda) \left(I_2(\mathbf{u}', \lambda) - L(\lambda) g_2(\mathbf{u}') S(\mathbf{u}, \lambda) \right) \\ &\quad + 2\rho_S \frac{S(\mathbf{u}, \lambda) - S_{\text{TV}}(\mathbf{u}, \lambda)}{\delta_S}. \end{aligned} \quad (10.19)$$

We approximate $\partial_{f(\mathbf{u})} W_2(\mathbf{u}')$ and $\partial_{f(\mathbf{u})} I_2(\mathbf{u}')$ with central finite differences that are pre-computed at the beginning of the optimization process. As we mentioned before, obtaining $\partial_{f(\mathbf{u})} g_2(\mathbf{u}')$ is not straightforward since we never optimize $g_2(\mathbf{u})$ in the regular lattice of the second image but only its representation warped to the first image through the flow $f(\mathbf{u})$. To overcome this limitation, we consider a linear approximation of its partial derivatives by extracting a neighborhood of points $\mathbf{u}_1 \dots \mathbf{u}_n$ in a square window centered at \mathbf{u} and computing a linear least square fit of the corresponding values $g_2(f(\mathbf{u}_1)), \dots, g_2(f(\mathbf{u}_n))$, resulting in a planar fit $g_2(f(\mathbf{u})) \approx c_0 + \mathbf{c}_1^T \mathbf{u}$. Then, the plane linear coefficient \mathbf{c}_1 is used as an approximation of the gradient $\partial_{f(\mathbf{u})} g_2(\mathbf{u}')$. With the gradient to hand, we update the estimation of the affine function hyper-parameters and the reflectance for each point by means of a gradient descent approach with a constant step size η .

Finally, for the third optimization step, we follow the fast iterative method proposed by Bresson and Chan [39].

10.3.1 Initialization

Note that our approach relies on an initial estimate of the flow $f(\mathbf{u})$, illuminant power spectrum and specular highlights. This is since, if the illuminant power spectrum and the specular coefficient is known, the reflectance and the shading factor can be obtained via algebraic manipulation and normalization operations [96]. Indeed, there are a number of methods elsewhere in the literature that can be used to obtain these initial estimates. Here, we use the method in [173] to recover the image highlights and that in [69] for the recovery of the initial estimate of the illuminant power spectrum.

For the optical flow, we avoid the common coarse-to-fine-approaches, proposing to rather exploit a small set of initial sparse matches as a starting point for the flow optimization. This is a similar approach to that used in recent works by Leordeanu *et al.* [114] or Brox and Malik [45] which are proven to deal with very large displacements.

To this end, we compute a small set of reliable sparse matches from an image pair following the method in [20] and making use of *SURF* features extracted from the initial shading factor. We modified the original pay-off function to include a similarity term that weights the angular error of the reflectance spectra among two matches. As a consequence, we are better able to select a good set of inliers without epipolar filtering, which is not a feasible option if the scene objects are allowed to move.

We use these sparse matches to get an initial estimate of the flow around a limited set of points in our optimization domain. We designed an energy functional composed by a data term and a simple $L2$ regularizer on the flow gradient.

$$E_{\text{flow}} = \alpha \int_{\Omega} D(\mathbf{u}) H(\mathbf{u}) [\mu_1 Es(\mathbf{u}) + \mu_2 Er(\mathbf{u})] + \|\partial_{\mathbf{u}} T(\mathbf{u})\|_2^2 d\mathbf{u} \quad (10.20)$$

with

$$Es(\mathbf{u}) = [g_1(\mathbf{u}) - g_2(\mathbf{u} + T(\mathbf{u}))]^2 \quad (10.21)$$

$$Er(\mathbf{u}) = e^{-\sum_{\lambda} S_1(\mathbf{u}, \lambda) S_2(\mathbf{u} + T(\mathbf{u}), \lambda)} \quad (10.22)$$

$$D(\mathbf{u}) = e^{-\frac{1}{\sigma} \min_{m \in M} \|\mathbf{u} - m\|} \quad (10.23)$$

$$H(\mathbf{u}) = \gamma \frac{\sum_{\lambda} \|\partial_{\mathbf{u}} S_1(\mathbf{u}, \lambda)\|^2}{\max_{\mathbf{u}'} \sum_{\lambda} \|\partial_{\mathbf{u}'} S_1(\mathbf{u}', \lambda)\|^2} + 1 \quad (10.24)$$

The data term accounts for both the photometric (Es) and material (Er) consistency between the two images through an $L2$ -norm penalty function. A spatial weighting term (D) modulate the effect of the regularizer with respect to the data term as a function of the distance from the closest match in the initial set M whereas (H) is used to allow discontinuities in the proximity of edges. By gradually changing the radius σ , we can expand the initial sparse support of the flow to the neighboring pixels. We treat minimization of such functional as a standard variational problem by solving the set of associated Euler-Lagrange equations [74].

10.3.2 Effect of the regularization terms

The Total Variation hyper-prior regularization term was introduced to enforce the patch-wise uniform material assumption and the locally uniform flow assumption formalized in terms of locally-affine transformation. Figure 10.1 shows the effect of the priors on the regularized parameters on a sample image pair. The top row shows the color image pair, the second row shows the reflectance as returned by [96] (H&RK) and as optimized by our process, the third row shows the gradient magnitude of the reflectances, while the last row shows the Forbenious norm of the differential of the hyper-parameters A at initialization and after optimization. It is immediately clear that the algorithm is capable of clustering together regions of uniform material that had significant variation in the estimated reflectance with H&RK. For example, look at the gradient magnitude of the reflectance in areas like the roof of the truck on the

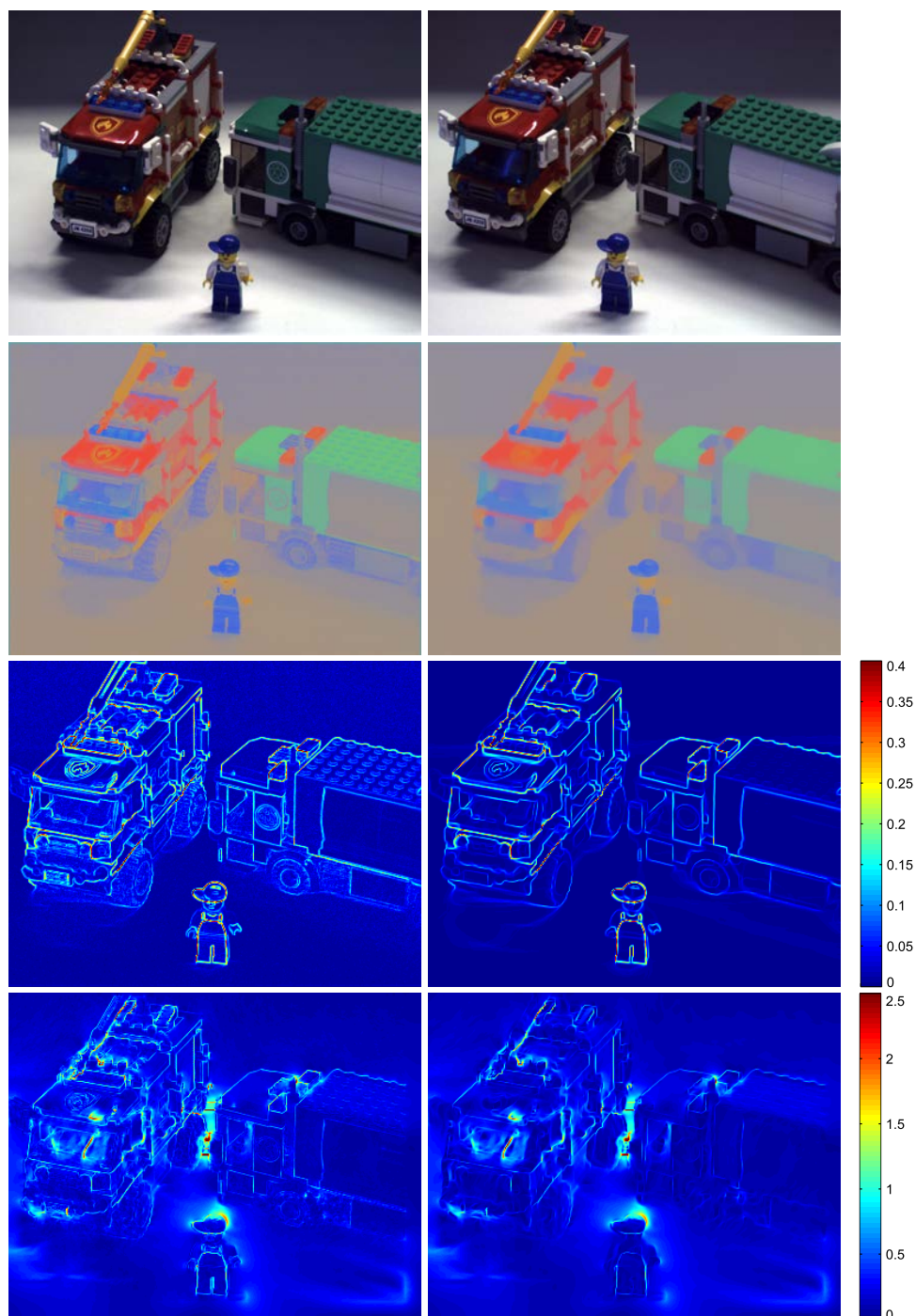


Figure 10.1: Sample image pair showing the effect of the priors on the regularized parameters. First row: input image pair. Second and third row: Reflectance value and gradient magnitude computed by H&RK (left) and by our approach (right). Last row: Frobenius norm of the differential of the hyper-parameters A at initialization and after optimization.

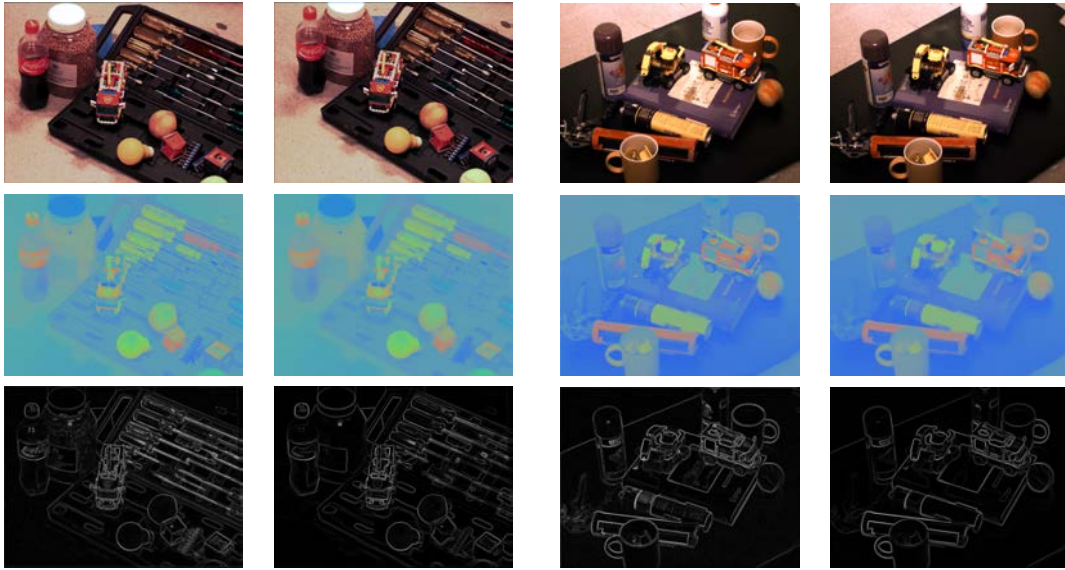


Figure 10.2: Reflectance obtained for two sample image pairs from scene 3 (left) and 4 (right). First row: input images. Second row: cromaticity obtained with H&RK and our method. Third row: gradient magnitude of the reflectances.

right or the wheels of the truck on the left. In both cases the materials are uniform and thus should exhibit uniform reflectance, but the wide variation in shading leaks into variations in the reflectance estimated with H&RK, on the other hand, our approach strongly reduces the variation in reflectance, while maintaining sharp variations across different materials. For the same reason, the regularization of the affine hyper-parameters significantly improve the details captured by the flow. Look for example at the flow around the logos on the two trucks: the logos correspond to a pure change in material, and should not have any effect on the flow, however, the edges of the logos are clearly visible in the gradient magnitude of the flow hyper-parameters at initialization, which indicates a leakage of information from the estimated reflectance to the estimated flow. After optimization, not only is the flow generally more uniform, with high gradient mostly in correspondence with depth discontinuities or occluded pixels, but the boundaries of the logos vanish almost completely.

10.4 Experiments

For purposes of comparison, we have used the method in [96]. Our choice hinges in the fact that the alternative is aimed at processing imaging spectroscopy data based upon the dichromatic model. Moreover, the method in [96] is an optimisation approach. Both our method and the alternative have been initialized using the same estimates of the illuminant power spectrum and specular highlights.

For our experiments, we have used four image sequences acquired using an uncal-



Figure 10.3: Shading (top-row) and specular factors (bottom-row) obtained for the same sample image pairs shown in Figure 10.2.

Table 10.1: RMS and Euclidean angular error for the illuminant recovered by our approach and the H&RK method for all the four scenes of our dataset.

Scene	H&RK Ang. Error	Our Ang. Error	H&RK RMS	Our RMS
1	0.080354	0.080045	0.026777	0.026675
2	0.066695	0.055120	0.023576	0.019485
3	0.076167	0.074565	0.025383	0.024849
4	0.021691	0.020638	0.007669	0.007296

ibrated multispectral camera delivering six channels in the visible spectrum and one in the near-infrared. It is worth noting in passing that our method can be easily applied to data comprising any number of wavelength bands, *i.e.* color, hyperspectral or multispectral. Each of our image sequences comprises 10 frames, depicting scenes containing a wide variety of objects made of different materials and depicting a wide variety of shapes. Each of these scenes is illuminated by different lights, spanning artificial sunlights, tungsten and incandescent lamps. For each of these, the illuminant power spectrum has been acquired using a LabSphere Spectralon calibration target. For our dataset, we have computed reflectance images for groundtruthing purposes following the procedure in [75].

In the top row of Figure 10.2 we show a pseudo-colour image pair for two sample scenes in our dataset. Colours are obtained in the same way as in [96]. In the second row, we show the initial (first and third column) and final (second and fourth columns) reflectance obtained by our method. The bottom row shows the gradient magnitude for the reflectance shown in the row above. Note that the total variation regularization process in our approach has improved the reflectance estimate by removing artifacts arising from the surface geometry.

Now we turn our attention to the shading factor and specular coefficient recovered by our method with respect to H&RK. Specifically, in Figure 10.3 we compare the

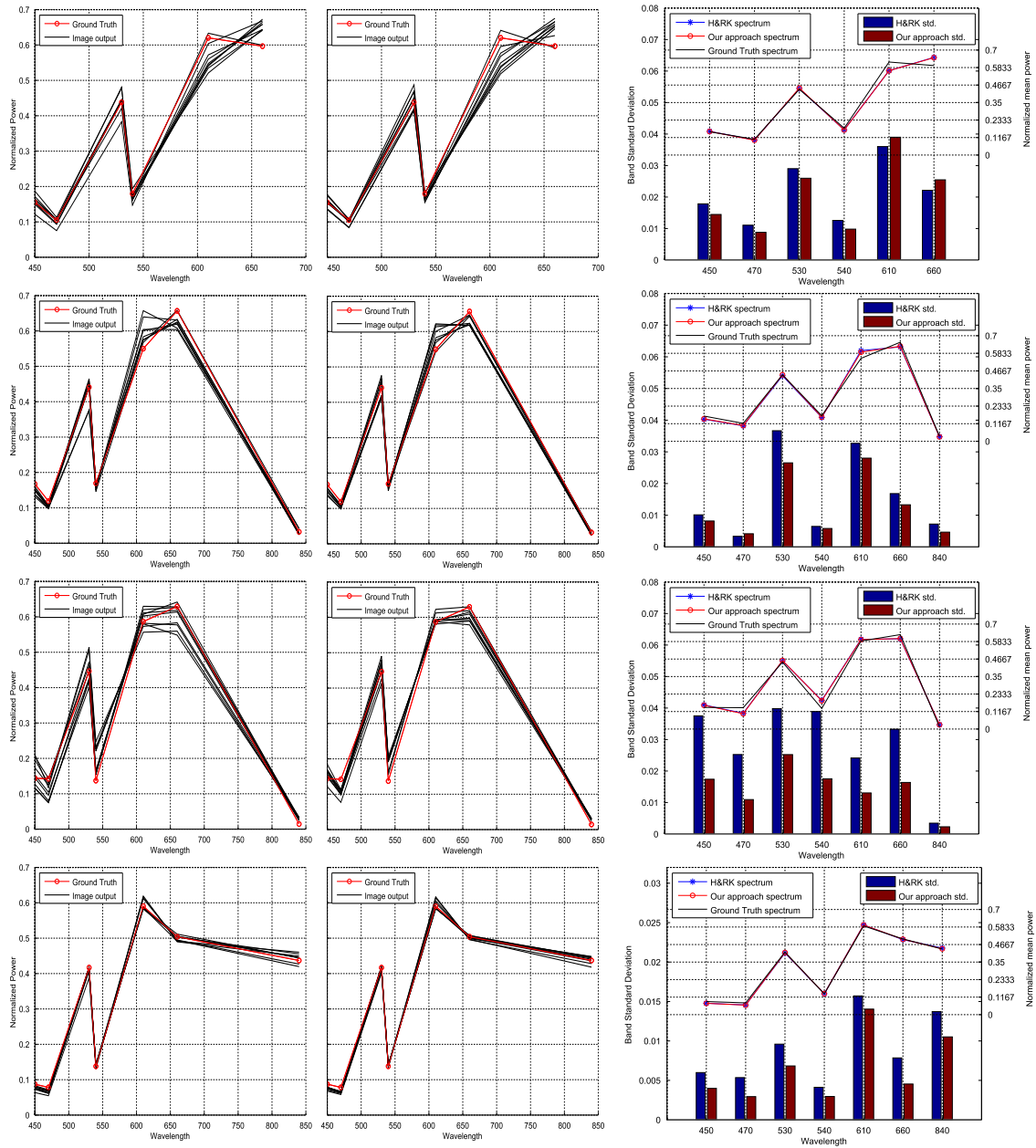


Figure 10.4: Illuminant power spectra for each scene. First and second column: power spectrum for each scene image as computed by H&RK and our approach respectively. Third column: average spectrum with the standard deviation for each band.

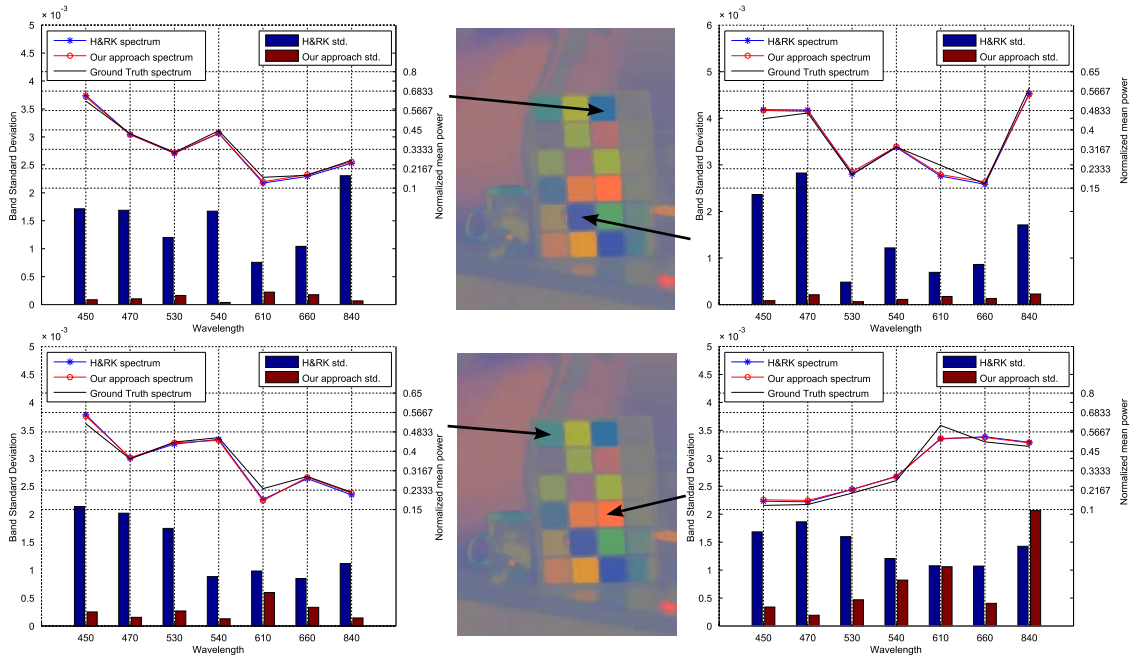


Figure 10.5: Reflectance spectra and the standard deviation for each band for the pixels inside the respective color tiles.

shading and specular maps delivered by the two approaches for the two scenes already analyzed in Figure 10.2. Note that the specular highlights delivered by our method are in better accordance with the input imagery as can be appreciated by observing the screwdriver box, the jars and the cups present in the scenes.

In Figures 10.4 and 10.5 we illustrate the results yielded by our method and the alternative regarding the recovery of the reflectance and the illuminant power spectrum. To this end, the left-hand side of Figure 10.4, we plot the illuminant delivered by H&RK and our method superimposed over the ground-truth (red-line). In the panel, the top trace depicts the spectrum whereas the bottom bar plot corresponds to the standard deviation of the illuminant per band over the corresponding image sequence. Note that the standard deviation for the illuminant power spectrum is much lower for our method. This is also the case for the reflectance. In Figure 10.5 we show the reflectance for four colour tiles on an XRite colour checker placed in one of our scenes. For the sake of clarity of presentation, the figure shows a close up of the color checker and, in

Table 10.2: Average and standard deviation of the RMS and Euclidean angular error of the estimated reflectance inside the colored tiles shown in Figure 10.5.

H&RK Ang. Error	Our Ang. Error	H&RK RMS	Our RMS
0.075242 ± 0.01740	0.073063 ± 0.02032	0.028431 ± 0.00657	0.027607 ± 0.00767

a fashion similar to Figure 10.4, the spectrum as a trace at the top of the plots with the standard deviation at the bottom on a bar plot.

In Table 10.1 we show the RMS and Euclidean angular error for the illuminant power spectrum recovered by our approach and the H&RK method across all the images for the four scenes in our dataset. Note that, for both measures, our method exhibits a lower error. This is consistent with our qualitative results showed earlier. Finally, in Table 10.2, we show the RMS and Euclidean angular error of the recovered reflectance averaged among each coloured tile shown in Figure 10.5.

10.5 Conclusions

In this chapter we proposed a novel method for dichromatic model recovery from an image pair by means of an energy minimization that simultaneously take into account the model parameters and the flow between the images. We introduced a novel affine hyper-prior for the flow that, in combination with a Total Variation regularization, provides a natural piecewise-rigid assumption on the motion under a weak camera model. The same kind of regularizer is used for the reflectance imposing the assumption that objects are composed by local patches of uniform materials. As a result, we are able to obtain a better reflectance estimation with respect to the current single-image state of the art approaches. Moreover, our approach has shown a significant lower variance while computing the illuminant spectrum over a sequence of images of the same scene. This behaviour is crucial for many applications for which a coherence of the dichromatic parameters is advisable when analyzing multiple instances of the same objects involved in a sequence for which we know that both the reflectance and the illuminant are constant. Furthermore, qualitative results shows that the method discriminates better between the shading (i.e. the geometrical features of a surface) and the texture an object.

11

Conclusions

In this thesis we approached different aspects of the camera calibration problem and presented two different scene acquisition applications.

In the first part of the thesis we covered different aspect of camera calibration using circular features. In Chapter 3 we presented a fiducial marker that exploits the interplay between different projective invariants to offer a simple, fast and accurate pose detection without requiring image rectification. Moreover, we tested the usage of such marker as a calibration target and exploited its free internal area to present an effective technique to calibrate a camera-projector setup. In Chapter 4 we presented a different type of fiducial marker which heavily relies on the robust framework of cyclic codes to offer superior occlusion resilience, accurate detection and robustness against various types of noise. We improved on the seminal version proposed in [31] by investigating their usage even for the uncalibrated case. Also in this case, the marker itself can be used as a calibration target with the advantage of being very robust against occlusions or mis-detections thus making it an excellent candidate to automatic calibration setups. Finally, in Chapter 5 we presented a method to recover the camera intrinsic parameters by exploiting the projective behaviour of a generic set of coplanar circles. Additionally, we developed a game-theoretic inlier selection process to discriminate a good cluster of coplanar circles from each image thus making our method particularly robust to be used in practice.

In the second part we moved our attention to a more powerful unconstrained camera model in which each single ray entering the camera is independently parametrized. In Chapter 6 we investigated the use of an unconstrained camera model to calibrate central quasi-pinhole cameras for high precision measurement and reconstruction tasks, and provided an effective approach to perform the calibration. Within the experimental section we showed how an unconstrained model can successfully eliminate the spatial coherence of the error, resulting in more precise and repeatable measures than what is achieved with the pinhole model. Furthermore, we studied and presented a valid interpolation function on the ray manifold to reconstruct the light field for any point in space. In Chapter 7 we exploited the calibration technique presented in the previous chapter to introduce an online projector calibration method based on the unconstrained imaging model that can be seamlessly applied to many commonly available 3D scanning systems. The advantage is an improved object re-

construction coverage and repeatability of the measures. Finally, in Chapter 8 we introduced a novel approach to lower the complexity of a complete unconstrained model toward a pinhole configuration but allowing a complete generic distortion map. The approach combines the simplicity of a central camera model, enabling the usage of powerful projective geometry tools (i.e. triangulation, vanishing point estimation, etc.), while sharing the ability of unconstrained models to accommodate non-standard lens characteristics. Moreover, the flexibility of the method allows to simultaneously estimate the rectification and undistortion map of a stereo rig.

In the last part of the thesis we presented two different scene acquisition applications. In Chapter 9 we described a novel approach for ellipse fitting that exploits multiple simultaneous calibrated views of the same physical elliptical object. The proposed method has been implemented in an industry-grade aluminum pipe intakes dimensional assessment tool. Furthermore, a complete GPU implementation allows to solve the level-set energy minimization problem with speedup improvements up to 10x. Finally, in Chapter 10 we proposed a novel method for dichromatic model recovery from an image pair by means of an energy minimization that simultaneously take into account the model parameters and the flow between the images. We exploit the usage of an affine hyper-prior combined with Total Variation regularization to provide a natural piecewise-rigid assumption on the motion under a weak camera model and to smooth the reflectance imposing the assumption that objects are composed by local patches of uniform materials.

11.1 Future Work

Most of the novel approaches presented in this thesis are not meant to be a complete exhaustive discussion on the topic. Conversely, we are already working on future possible improvements. For camera calibration topic, we demonstrated in this thesis how effective and general can be the usage of a totally unconstrained (raxel-based) model for any type of camera. We are working to improve the seminal work presented here in various ways.

First, we are studying a method to calibrate a structured-light system composed by just one camera and a projector. This would be substantially different from the case presented in Chapter 7 in which we assumed to have a stereo rig. Indeed, the method presented in Chapter 6 cannot be applied as is because the projector cannot “acquire” any information from a known target. Nevertheless, by exploiting the planarity of a surface, the fact that we know exactly the observed code (i.e. projected) from each pixel of the projector and a pre-calibration of the camera we are already obtaining some preliminary encouraging results.

Second, we have already tested our rays calibration method on a Lytro™ plenoptic camera with very promising results. Therefore, we are experimenting our proposed rays interpolation approach to determine if the current state-of-the-art performance on camera refocusing can be improved. Furthermore, we would like to exploit our ray-

based model to simultaneously estimate the micro-motion of a plenoptic camera and the de-blurring kernel to improve the quality of images taken from long exposures. The rationale is the following: there already exists well established methods in literature to revert the motion blur of a moving camera to restore the focus of the image. However, they rely on a good estimation of the blur kernel that can be estimated by knowing the motion performed by the camera during the exposure. But, if the camera is not central, our guess is that small motions can be estimated with far more accuracy since we effectively can observe the same scene point from multiple points of view.

Third, we would like to test our non-parametric lens distortion approach in case of color cameras. In this case, if we perform the calibration on different channels (i.e. selected pixels on the CCD due to the Bayer filter) we should obtain slightly different undistortion maps when the mounted lens suffers from color aberration. Therefore, by computing the undistortion map for each channel, it would be possible to completely remove the color aberration for any type of lens.

For the topic of optical flow and dichromatic parameter recovery we are working on improving the formulation (and the total variation regularizer) to use a complete homography as an high-order smoothing prior. This would have the physical meaning of assuming the image composed by locally planar patches. However, this approach would require to leverage the TV optimization from a vector field to a manifold (i.e. 3×3 matrices constrained for example by a unitary Frobenius norm). Moreover, this approach would require a valid initialization of the supposed homography transformation for each pixel. We are experimenting different alternatives that exploit the camera motion recovered by common structure-from-motion techniques.

Bibliography

- [15] ADELSON, E., AND WANG, J. Single lens stereo with a plenoptic camera. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 14, 2 (Feb 1992), 99–106.
- [16] ADELSON, E. H., AND BERGEN, J. R. The plenoptic function and the elements of early vision. In *Computational Models of Visual Processing* (1991), MIT Press, pp. 3–20.
- [17] ALBARELLI, A., BERGAMASCO, F., AND TORSSELLO, A. Rigid and non-rigid shape matching for mechanical components retrieval. In *Computer Information Systems and Industrial Management*, A. Cortesi, N. Chaki, K. Saeed, and S. Wierzchoń, Eds., vol. 7564 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2012, pp. 168–179.
- [18] ALBARELLI, A., BULÒ, S. R., TORSSELLO, A., AND PELILLO, M. Matching as a non-cooperative game. In *ICCV: IEEE Intl. Conf. on Comp. Vis.* (2009), IEEE Computer Society.
- [19] ALBARELLI, A., RODOLÀ, E., AND TORSSELLO, A. Robust camera calibration using inaccurate targets. In *Proceedings of the British Machine Vision Conference* (2010), BMVA Press, pp. 16.1–16.10. doi:10.5244/C.24.16.
- [20] ALBARELLI, A., RODOLÀ, E., AND TORSSELLO, A. Imposing semi-local geometric constraints for accurate correspondences selection in structure from motion: A game-theoretic perspective. *IJCV* 97, 1 (2012), 36–53.
- [21] ALISMAIL, H. S., BROWNING, B., AND DIAS, M. B. Evaluating pose estimation methods for stereo visual odometry on robots. In *the 11th International Conference on Intelligent Autonomous Systems (IAS-11)* (2011).
- [22] AMELLER, M.-A., TRIGGS, B., AND QUAN, L. Camera Pose Revisited – New Linear Algorithms, 2000. Submitted to ECCV’00 Submitted to ECCV’00.
- [23] ANONYMOUS. Anonymized for double blind review.
- [24] ARUN, K., HUANG, T. S., AND BLOSTEIN, S. D. Least-squares fitting of two 3-d point sets. *Pattern Analysis and Machine Intelligence, IEEE Transactions on PAMI-9*, 5 (Sept 1987), 698–700.
- [25] AUDET, S., AND OKUTOMI, M. A user-friendly method to geometrically calibrate projector-camera systems. In *Computer Vision and Pattern Recognition Workshops, 2009. CVPR Workshops 2009. IEEE Computer Society Conference on* (2009), pp. 47–54.

- [26] BAKER, S., AND MATTHEWS, I. Lucas-kanade 20 years on: A unifying framework. *Int. J. Comput. Vision* 56, 3 (Feb. 2004), 221–255.
- [27] BAKER, S., SCHARSTEIN, D., LEWIS, J., ROTH, S., BLACK, M., AND SZELISKI, R. A database and evaluation methodology for optical flow. *International Journal of Computer Vision* 92, 1 (2011), 1–31.
- [28] BARNARD, K., MARTIN, L., AND FUNT, B. V. Colour by Correlation in a Three-Dimensional Colour Space. In *European Conference on Computer Vision* (2000), pp. 375–389.
- [29] BARRON, J. L., FLEET, D. J., AND BEAUCHEMIN, S. S. Performance of optical flow techniques. *Int. Journal of Computer Vision* 12, 1 (1994), 43–77.
- [30] BENETAZZO, A., BERGAMASCO, F., BARBARIOL, F., TORSSELLO, A., CARNIEL, S., AND SCLAVO, M. Toward an operational stereo system for directional wave measurements from moving platforms. In *OMAE 2014. International Conference on Ocean, Offshore and Artic Engineering* (8 2014-jun. 13 2014).
- [31] BERGAMASCO, F., ALBARELLI, A., RODOLA, E., AND TORSSELLO, A. Rune-tag: A high accuracy fiducial marker with strong occlusion resilience. In *Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition* (Washington, DC, USA, 2011), CVPR '11, IEEE Computer Society, pp. 113–120.
- [32] BERGAMASCO, F., ALBARELLI, A., RODOLA, E., AND TORSSELLO, A. Can a fully unconstrained imaging model be applied effectively to central cameras? In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on* (June 2013), pp. 1391–1398.
- [33] BESL, P. J., AND MCKAY, N. D. A method for registration of 3-D shapes. *IEEE Trans. Pattern Anal. Mach. Intell.* 14, 2 (1992), 239–256.
- [34] BI, Z., AND WANG, L. Advances in 3d data acquisition and processing for industrial applications. *Robotics and Computer-Integrated Manufacturing* 26, 5 (2010), 403 – 413.
- [35] BOOKSTEIN, F. L. Fitting conic sections to scattered data. *Computer Graphics and Image Processing* 9, 1 (1979), 56 – 71.
- [36] BRADSKI, G., AND KAEHLER, A. *Learning OpenCV: Computer Vision with the OpenCV Library*, 1st ed. O'Reilly Media, Inc., 2008.
- [37] BRAINARD, D. H., DELAHUNT, P. B., FREEMAN, W. T., KRAFT, J. M., AND XIAO, B. Bayesian model of human color constancy. *Journal of Vision* 6, 11 (2006), 1267–1281.

- [38] BRELSTAFF, G., AND BLAKE, A. Detecting specular reflection using lambertian constraints. In *Int. Conference on Computer Vision* (1988), pp. 297–302.
- [39] BRESSON, X., AND CHAN, T. F. Fast dual minimization of the vectorial total variation norm and applications to color image processing, 2008.
- [40] BROWN, D. C. Decentering Distortion of Lenses. *Photometric Engineering* 32, 3 (1966), 444–462.
- [41] BROWN, D. C. Close-range camera calibration. *Photogrammetric Engineering* 37, 8 (1971), 855–866.
- [42] BROWN, D. C. Close-range camera calibration. *PHOTOGRAMMETRIC ENGINEERING* 37, 8 (1971), 855–866.
- [43] BROX, T., BRUHN, A., PAPENBERG, N., AND WEICKERT, J. High accuracy optical flow estimation based on a theory for warping. In *Computer Vision - ECCV 2004*, T. Pajdla and J. Matas, Eds., vol. 3024 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2004, pp. 25–36.
- [44] BROX, T., AND MALIK, J. Large displacement optical flow: Descriptor matching in variational motion estimation. *IEEE Trans. Pattern Anal. Mach. Intell.* 33, 3 (Mar. 2011), 500–513.
- [45] BROX, T., AND MALIK, J. Large displacement optical flow: Descriptor matching in variational motion estimation. *IEEE TPAMI* 33, 3 (March 2011), 500–513.
- [46] CABRERA, J., AND MEER, P. Unbiased estimation of ellipses by bootstrapping. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 18, 7 (Jul 1996), 752–756.
- [47] CAMERON, J., AND LASENBY, J. Estimating human skeleton parameters and configuration in real-time from marked optical motion capture. In *Conference on Articulated Motion and Deformable Objects* (2008).
- [48] CAMPBELL, J., SUKTHANKAR, R., AND NOURBAKHSI, I. Techniques for evaluating optical flow for visual odometry in extreme terrain. In *Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on* (Sept 2004), vol. 4, pp. 3704–3711 vol.4.
- [49] CHEN, C.-Y., AND CHIEN, H.-J. An incremental target-adapted strategy for active geometric calibration of projector-camera systems. *Sensors* 13, 2 (2013), 2664–2681.
- [50] CHEN, Q., WU, H., AND WADA, T. Camera calibration with two arbitrary coplanar circles. In *European Conference on Computer Vision - ECCV* (2004).

- [51] CHEN, X., XI, J., JIN, Y., AND SUN, J. Accurate calibration for a camera-projector measurement system based on structured light projection. *Optics and Lasers in Engineering* 47, 3-4 (2009), 310–319.
- [52] CHEN, Y., AND IP, H. H. S. Planar metric rectification by algebraically estimating the image of the absolute conic. In *Proceedings of the Pattern Recognition, 17th International Conference on (ICPR'04) Volume 4 - Volume 04* (Washington, DC, USA, 2004), ICPR '04, IEEE Computer Society, pp. 88–91.
- [53] CHIA, A., LEUNG, M., ENG, H.-L., AND RAHARDJA, S. Ellipse detection with hough transform in one dimensional parametric space. In *Image Processing, 2007. ICIP 2007. IEEE International Conference on* (16 2007-oct. 19 2007), vol. 5, pp. V–333–V–336.
- [54] CHO, Y., LEE, J., AND NEUMANN, U. A multi-ring color fiducial system and a rule-based detection method for scalable fiducial-tracking augmented reality. In *Proceedings of International Workshop on Augmented Reality* (1998).
- [55] CLAUS, D., AND FITZGIBBON, A. W. A rational function lens distortion model for general cameras. In *Proc. IEEE Computer Vision and Pattern Recognition* (2005), pp. 213–219.
- [56] CLAUS, D., AND FITZGIBBON, A. W. Reliable automatic calibration of a marker-based position tracking system. In *IEEE Workshop on Applications of Computer Vision* (2005).
- [57] COLOMBO, C., COM, D., AND BIMBO, A. D. Camera calibration with two arbitrary coaxial circles. In *In Proc. 9th European Conference on Computer Vision ECCV 2006* (2006), Springer, pp. 265–276.
- [58] COOPER, A. Finding bch error locator polynomials in one step. *Electronics Letters* 27, 22 (Oct 1991), 2090–2091.
- [59] COXETER, H. S. M. *Projective Geometry, 2nd ed.* Springer Verlag, 2003.
- [60] DAVISON, A. J., REID, I. D., MOLTON, N. D., AND STASSE, O. Monoslam: Real-time single camera slam. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26, 6 (2007), 1052–1067.
- [61] DEVERNAY, F., AND FAUGERAS, O. Straight lines have to be straight: Automatic calibration and removal of distortion from scenes of structured environments. *Mach. Vision Appl.* 13, 1 (Aug. 2001), 14–24.
- [62] DOUXCHAMPS, D., AND CHIHARA, K. High-accuracy and robust localization of large control markers for geometric camera calibration. *IEEE Trans. on Pattern Analysis and Machine Intell.* 31 (2009), 376–383.

- [63] DRULEA, M., AND NEDEVSCHI, S. Total variation regularization of local-global optical flow. In *Intelligent Transportation Systems (ITSC), 2011 14th International IEEE Conference on* (Oct 2011), pp. 318–323.
- [64] DUFOURNAUD, Y., HORAUD, R., AND QUAN, L. Robot Stereo-hand Coordination for Grasping Curved Parts. In *9th British Machine Vision Conference (BMVC '98)* (Southampton, Royaume-Uni, 1998), J. N. Carter and M. S. Nixon, Eds., vol. 2, British Machine Vision Association, pp. 760–769.
- [65] FAUGERAS, O. *Three-dimensional Computer Vision: A Geometric Viewpoint*. MIT Press, Cambridge, MA, USA, 1993.
- [66] FIALA, M. Linear markers for robot navigation with panoramic vision. In *Proc. of the 1st Canadian Conf. on Computer and Robot Vision* (Washington, DC, USA, 2004), CRV '04, IEEE Computer Society, pp. 145–154.
- [67] FIALA, M. Designing highly reliable fiducial markers. *IEEE Trans. Pattern Anal. Mach. Intel.* 32, 7 (2010).
- [68] FINLAYSON, G. D., AND SCHAEFER, G. Convex and non-convex illuminant constraints for dichromatic colour constancy. In *IEEE CVPR* (2001), pp. 1:598–604.
- [69] FINLAYSON, G. D., AND SCHAEFER, G. Solving for colour constancy using a constrained dichromatic reflection model. *IJCV* 42, 3 (2001), 127–144.
- [70] FISS, J., CURLESS, B., AND SZELISKI, R. Refocusing plenoptic images using depth-adaptive splatting. *2014 IEEE International Conference on Computational Photography (ICCP) 0* (2014), 1–9.
- [71] FITZGIBBON, A., PILU, M., AND FISHER, R. Direct least square fitting of ellipses. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 21, 5 (may 1999), 476–480.
- [72] FITZGIBBON, A., PILU, M., AND FISHER, R. B. Direct least square fitting of ellipses. *IEEE Trans. Pattern Anal. Mach. Intell.* 21, 5 (May 1999), 476–480.
- [73] FOFI, D., SALVI, J., AND MOUADDIB, E. Uncalibrated vision based on structured light. In *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on* (2001), vol. 4, pp. 3548–3553.
- [74] FORSYTH, A. *Calculus of variations*. Dover books on advanced mathematics. Dover Publications, 1960.
- [75] FOSTER, D. H., AMANO, K., NASCIMENTO, S. M. C., AND FOSTER, M. J. Frequency of metamerism in natural scenes. *J. Opt. Soc. America A* 23, 10 (2006), 2359–2372.

- [76] FURUKAWA, R., AND KAWASAKI, H. Dense 3d reconstruction with an uncalibrated stereo system using coded structured light. In *Computer Vision and Pattern Recognition - Workshops, 2005. CVPR Workshops. IEEE Computer Society Conference on* (2005), pp. 107–107.
- [77] GATRELL, L., HOFF, W., AND SKLAIR, C. Robust image features: Concentric contrasting circles and their image extraction. In *Proc. of Cooperative Intelligent Robotics in Space* (Washington, USA, 1991), SPIE.
- [78] GEORGIEV, T., YU, Z., LUMSDAINE, A., AND GOMA, S. Lytro camera technology: theory, algorithms, performance analysis, 2013.
- [79] GREINER, W. *Quantum mechanics: an introduction; 3rd ed.* Springer, Berlin, 1994. Includes examples.
- [80] GROSSBERG, M. D., AND NAYAR, S. K. A general imaging model and a method for finding its parameters. In *International Conference of Computer Vision* (2001), pp. 108–115.
- [81] GUPTA, A., LITTLE, J., AND WOODHAM, R. Using line and ellipse features for rectification of broadcast hockey video. In *Computer and Robot Vision (CRV), 2011 Canadian Conference on* (may 2011), pp. 32–39.
- [82] HANSEN, D. W., AND JI, Q. In the eye of the beholder: A survey of models for eyes and gaze. *IEEE Trans. Pattern Anal. Mach. Intell.* 32, 3 (Mar. 2010), 478–500.
- [83] HARTLEY, R., GUPTA, R., AND CHANG, T. Stereo from uncalibrated cameras. In *Computer Vision and Pattern Recognition, 1992. Proceedings CVPR '92., 1992 IEEE Computer Society Conference on* (1992), pp. 761–764.
- [84] HARTLEY, R., AND KANG, S. B. Parameter-free radial distortion correction with centre of distortion estimation. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on* (Oct 2005), vol. 2, pp. 1834–1841 Vol. 2.
- [85] HARTLEY, R., AND ZISSERMAN, A. *Multiple View Geometry in Computer Vision*, 2 ed. Cambridge University Press, New York, NY, USA, 2003.
- [86] HARTLEY, R. I., AND ZISSERMAN, A. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.
- [87] HEIKKILÄ, J. Moment and curvature preserving technique for accurate ellipse boundary detection. In *Pattern Recognition, 1998. Proceedings. Fourteenth International Conference on* (aug 1998), vol. 1, pp. 734–737 vol.1.
- [88] HEIKKILÄ, J. Geometric camera calibration using circular control points. *IEEE Trans. Pattern Anal. Mach. Intell.* 22 (October 2000), 1066–1077.

- [89] HORN, B. K. *Robot Vision*, 1st ed. McGraw-Hill Higher Education, 1986.
- [90] HORN, B. K. P. Closed-form solution of absolute orientation using unit quaternions. *Journal of the Optical Society of America A* 4, 4 (1987), 629–642.
- [91] HORN, B. K. P. Closed-form solution of absolute orientation using unit quaternions. *J. of the Optical Society of America. A* 4, 4 (Apr 1987), 629–642.
- [92] HORN, B. K. P., HILDEN, H., AND NEGAHDARIPOUR, S. Closed-form solution of absolute orientation using orthonormal matrices. *JOURNAL OF THE OPTICAL SOCIETY AMERICA* 5, 7 (1988), 1127–1135.
- [93] HORN, B. K. P., AND SCHUNCK, B. G. Determining optical flow. *ARTIFICIAL INTELLIGENCE* 17 (1981), 185–203.
- [94] HU, H., FERNANDEZ-STEEGER, T., DONG, M., NGUYEN, H. T., AND AZZAM, R. 3d modeling using lidar data and its geological and geotechnical applications. In *Geoinformatics, 2010 18th International Conference on* (June 2010), pp. 1–6.
- [95] HUANG, J., WANG, Z., GAO, J., AND XUE, Q. Projector calibration with error surface compensation method in the structured light three-dimensional measurement system. *Optical Engineering* 52, 4 (2013), 043602–043602.
- [96] HUYNH, C. P., AND ROBLES-KELLY, A. A solution of the dichromatic model for multispectral photometric invariance. *IJCV* 90, 1 (2010), 1–27.
- [97] HUYNH, C. P., ROBLES-KELLY, A., AND HANCOCK, E. R. Shape and refractive index recovery from single-view polarisation images. In *IEEE CVPR* (2010).
- [98] HUYNH, D. Q. The cross ratio: A revisit to its probability density function. In *Proceedings of the British Machine Vision Conference BMVC 2000* (2000).
- [99] JIANG, G., AND QUAN, L. Detection of concentric circles for camera calibration. *Computer Vision, IEEE International Conference on* 1 (2005), 333–340.
- [100] KANATANI, K. Ellipse fitting with hyperaccuracy. In *Computer Vision – ECCV 2006*, A. Leonardis, H. Bischof, and A. Pinz, Eds., vol. 3951 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2006, pp. 484–495.
- [101] KANNALA, J., SALO, M., AND HEIKKILÄ, J. Algorithms for computing a planar homography from conics in correspondence. In *British Machine Vision Conference* (2006).
- [102] KARIYA, T., AND KURATA, H. *Generalized Least Squares*. Wiley, 2004.

- [103] KATO, H., AND BILLINGHURST, M. Marker tracking and hmd calibration for a video-based augmented reality conferencing system. In *Proc. of the 2nd IEEE and ACM International Workshop on Augmented Reality* (Washington, DC, USA, 1999), IEEE Computer Society.
- [104] KAVAN, L., COLLINS, S., O’SULLIVAN, C., AND ŽÁRA, J. Dual quaternions for rigid transformation blending. Tech. Rep. TCD-CS-2006-46, Trinity College Dublin, 2006.
- [105] KAZHDAN, M., BOLITHO, M., AND HOPPE, H. Poisson surface reconstruction. In *Proceedings of the fourth Eurographics symposium on Geometry processing* (Aire-la-Ville, Switzerland, Switzerland, 2006), SGP ’06, pp. 61–70.
- [106] KIMURA, M., MOCHIMARU, M., AND KANADE, T. Projector calibration using arbitrary planes and calibrated camera. In *Computer Vision and Pattern Recognition, 2007. CVPR ’07. IEEE Conference on* (2007), pp. 1–8.
- [107] KLAUS, AND DORFMÜLLER. Robust tracking for augmented reality using retroreflective markers. *Computers and Graphics* 23, 6 (1999), 795 – 800.
- [108] KLINKER, G., SHAFER, S., AND KANADE, T. A physical approach to color image understanding. *Intl. Journal of Computer Vision* 4, 1 (1990), 7–38.
- [109] KNYAZ, V. A., GROUP, H. O., AND SIBIRYAKOV, R. V. The development of new coded targets for automated point identification and non-contact surface measurements. In *3D Surface Measurements, International Archives of Photogrammetry and Remote Sensing* (1998).
- [110] LAMBERT, H. H. *Photometria, sive De mensura et gradibus luminus, colorum et umbrae*, 1st ed. Augsburg: Eberhard Klett, 1760.
- [111] LAND, E. H. Recent advances in retinex theory. *Vision Research* 26, 1 (1986), 7–21.
- [112] LAND, E. H., AND MCCANN, J. J. Lightness and retinex theory. *J. Opt. Soc. Am* 61 (1971), 1–11.
- [113] LEE, J. C., DIETZ, P. H., MAYNES-AMINZADE, D., RASKAR, R., AND HUDSON, S. E. Automatic projector calibration with embedded light sensors. In *Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology* (New York, NY, USA, 2004), UIST ’04, ACM, pp. 123–126.
- [114] LEORDEANU, M., ZANFIR, A., AND SMINCHISESCU, C. Locally affine sparse-to-dense matching for motion and occlusion estimation. In *IEEE ICCV* (December 2013).

- [115] LEPETIT, V., MORENO-NOGUER, F., AND FUA, P. Epanp: An accurate $o(n)$ solution to the pnp problem. *International Journal of Computer Vision* 81, 2 (2009), 155–166.
- [116] LI, Y., WANG, Y.-T., AND LIU, Y. Fiducial marker based on projective invariant for augmented reality. *Journal of Computer Science and Technology* 22 (2007), 890–897.
- [117] LIERE, R. V., AND MULDER, J. D. Optical tracking using projective invariant marker pattern properties. In *Proceedings of the IEEE Virtual Reality Conference* (2003), IEEE Press.
- [118] LILIENBLUM, E., AND MICHAELIS, B. Optical 3d surface reconstruction by a multi-period phase shift method. *JCP* 2, 2 (2007), 73–83.
- [119] LIN, S., AND SHUM, H. Separation of diffuse and specular reflection in color images. In *Int. Conf. on Comp. Vision and Patt. Recognition* (2001).
- [120] LINT, J. H. V. *Introduction to Coding Theory*, 3rd ed. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1998.
- [121] LIU, C., YUEN, J., AND TORRALBA, A. Sift flow: Dense correspondence across scenes and its applications. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 33, 5 (May 2011), 978–994.
- [122] LOAIZA, M., RAPOSO, A., AND GATTASS, M. A novel optical tracking algorithm for point-based projective invariant marker patterns. In *Proceedings of the 3rd international conference on Advances in visual computing - Volume Part I* (Berlin, Heidelberg, 2007), ISVC'07, Springer-Verlag, pp. 160–169.
- [123] LU, C.-P., HAGER, G., AND MJOLSNESS, E. Fast and globally convergent pose estimation from video images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 22, 6 (Jun 2000), 610–622.
- [124] MA, L., HEN, Y., AND MOORE, K. L. Flexible camera calibration using a new analytical radial undistortion formula with application to mobile robot localization. In *IEEE International Symposium on Intelligent Control* (2003), pp. 799–804.
- [125] MACWILLIAMS, F., AND SLOANE, N. *The Theory of Error-Correcting Codes*, 2nd ed. North-holland Publishing Company, 1978.
- [126] MAIDI, M., DIDIER, J.-Y., ABABSA, F., AND MALLEM, M. A performance study for camera pose estimation using visual marker based tracking. *Mach. Vision Appl.* 21 (2010).

- [127] MALAMAS, E. N., PETRAKIS, E. G., ZERVAKIS, M., PETIT, L., AND LEGAT, J.-D. A survey on industrial vision systems, applications and tools. *Image and Vision Computing* 21, 2 (2003), 171 – 188.
- [128] MALASSIOTIS, S., AND STRINTZIS, M. Stereo vision system for precision dimensional inspection of 3d holes. *Machine Vision and Applications* 15 (2003), 101–113.
- [129] MALLON, J., AND WHELAN, P. F. Which pattern? biasing aspects of planar calibration patterns and detection methods. *Pattern Recognition Letters* 28, 8 (2007), 921 – 930.
- [130] MARTELLI, S., MARZOTTO, R., COLOMBARI, A., AND MURINO, V. Fpga-based robust ellipse estimation for circular road sign detection. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on* (june 2010), pp. 53 –60.
- [131] MAYBANK, S., AND FAUGERAS, O. A theory of self-calibration of a moving camera. *International Journal of Computer Vision* 8, 2 (1992), 123–151.
- [132] MCLAUGHLIN, R. A. Randomized hough transform: Improved ellipse detection with comparison. *Pattern Recognition Letters* 19, 3–4 (1998), 299 – 305.
- [133] MEDIONI, G., AND KANG, S. B. *Emerging Topics in Computer Vision*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 2004.
- [134] MEER, P., LENZ, R., AND RAMAKRISHNA, S. Efficient invariant representations. *Int. J. Comput. Vision* 26 (1998), 137–152.
- [135] MENG, X., AND HU, Z. A new easy camera calibration technique based on circular points. *Pattern Recognition* 36, 5 (2003), 1155 – 1164.
- [136] MIYAMOTO, K. Fish eye lens. *J. Opt. Soc. Am.* 54, 8 (Aug 1964), 1060–1061.
- [137] MORENO, D., AND TAUBIN, G. Simple, accurate, and robust projector-camera calibration. In *3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT), 2012 Second International Conference on* (2012), pp. 464–471.
- [138] MORÉ, J. The levenberg-marquardt algorithm: Implementation and theory. In *Numerical Analysis*, G. Watson, Ed., vol. 630 of *Lecture Notes in Mathematics*. Springer Berlin Heidelberg, 1978, pp. 105–116.
- [139] MUKHERJEE, K., AND MUKHERJEE, A. Joint optical flow motion compensation and video compression using hybrid vector quantization. In *Proceedings of the Conference on Data Compression* (Washington, DC, USA, 1999), DCC '99, IEEE Computer Society, pp. 541–.

- [140] NAIMARK, L., AND FOXLIN, E. Circular data matrix fiducial system and robust image processing for a wearable vision-inertial self-tracker. In *Proceedings of the 1st Int. Symposium on Mixed and Augmented Reality* (Washington, DC, USA, 2002), ISMAR '02, IEEE Computer Society.
- [141] NARASIMHAN, S. G., AND NAYAR, S. K. Contrast restoration of weather degraded images. *IEEE TPAMI* 25 (2003), 713–724.
- [142] NAYAR, S., AND BAKER, S. Catadioptric Image Formation. In *DARPA Image Understanding Workshop (IUW)* (May 1997), pp. 1431–1438.
- [143] NAYAR, S., AND BOLLE, R. Reflectance based object recognition. *International Journal of Computer Vision* 17, 3 (1996), 219–240.
- [144] NGUYEN, T. M., AHUJA, S., AND WU, Q. M. J. A real-time ellipse detection based on edge grouping. In *Systems, Man and Cybernetics, 2009. SMC 2009. IEEE International Conference on* (2009), pp. 3280–3286.
- [145] OKATANI, T., AND DEGUCHI, K. Autocalibration of a projector-camera system. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27, 12 (2005), 1845–1855.
- [146] OUELLET, J., AND HEBERT, P. A simple operator for very precise estimation of ellipses. In *Computer and Robot Vision, 2007. CRV '07. Fourth Canadian Conference on* (May 2007), pp. 21–28.
- [147] OUELLET, J., AND HEBERT, P. Precise ellipse estimation without contour point extraction. *Mach. Vision Appl.* 21 (2009).
- [148] POCK, T., CREMERS, D., BISCHOF, H., AND CHAMBOLLE, A. An algorithm for minimizing the mumford-shah functional. In *ICCV* (2009), pp. 1133–1140.
- [149] QUAN, L. Conic reconstruction and correspondence from two views. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 18, 2 (feb 1996), 151 – 160.
- [150] QUAN, L., AND LAN, Z. Linear n-point camera pose determination. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 21, 8 (Aug 1999), 774–780.
- [151] RAMALINGAM, S., STURM, P., AND LODHA, S. K. Towards complete generic camera calibration. In *Proc. IEEE Computer Vision and Pattern Recognition* (2005), pp. 1093–1098.
- [152] REID, G., TANG, J., AND ZHI, L. A complete symbolic-numeric linear method for camera pose determination. In *Proceedings of the 2003 International Symposium on Symbolic and Algebraic Computation* (New York, NY, USA, 2003), ISSAC '03, ACM, pp. 215–223.

- [153] ROBLES-KELLY, A., AND HUYNH, C. P. *Imaging Spectroscopy for Scene Analysis*. Springer, 2013.
- [154] RODOLÀ, E., ALBARELLI, A., BERGAMASCO, F., AND TORSSELLO, A. A scale independent selection process for 3d object recognition in cluttered scenes. *International Journal of Computer Vision* 102, 1-3 (2013), 129–145.
- [155] RUDIN, L. I., OSHER, S., AND FATEMI, E. Nonlinear total variation based noise removal algorithms. *Phys. D* 60, 1-4 (Nov. 1992), 259–268.
- [156] RUSINKIEWICZ, S., AND LEVOY, M. Efficient variants of the icp algorithm. In *Proc. of the Third Intl. Conf. on 3D Digital Imaging and Modeling* (2001), pp. 145–152.
- [157] SALVI, J., BATLLE, J., AND MOUADDIB, E. A robust-coded pattern projection for dynamic 3d scene measurement. *Pattern Recognition Letters* 19, 11 (1998), 1055 – 1065.
- [158] SAUVOLA, J., AND PIETIKAINEN, M. Adaptive document image binarization. *Pattern Recognition* 33, 2 (2000), 225 – 236.
- [159] SCHIKORA, M., KOCH, W., AND CREMERS, D. Multi-object tracking via high accuracy optical flow and finite set statistics. In *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on* (May 2011), pp. 1409–1412.
- [160] SCHLEICHER, D., AND ZAGAR, B. Image processing to estimate the ellipticity of steel coils using a concentric ellipse fitting algorithm. In *Signal Processing, 2008. ICSP 2008. 9th International Conference on* (oct. 2008), pp. 884 –890.
- [161] SCHMALZ, C., FORSTER, F., AND ANGELOPOULOU, E. Camera calibration: active versus passive targets. *Optical Engineering* 50, 11 (2011).
- [162] SHAFER, S. A. Using color to separate reflection components. *Color Research and Applications* 10, 4 (1985), 210–218.
- [163] SHAFER, S. A. Color. Jones and Bartlett Publishers, Inc., USA, 1992, ch. Using Color to Separate Reflection Components, pp. 43–51.
- [164] SHAH, S., AND AGGARWAL, J. Intrinsic parameter calibration procedure for a (high-distortion) fish-eye lens camera with distortion model and accuracy estimation*. *Pattern Recognition* 29, 11 (1996), 1775 – 1788.
- [165] SOCIETY FOR PHOTOGRAMMETRY, A., REMOTE SENSING, SLAMA, C. C., THEURER, C., AND HENRIKSEN, S. W., Eds. *Manual of photogrammetry*. Falls Church, Va. American Society of Photogrammetry, 1980.

- [166] SONG, G., AND WANG, H. A fast and robust ellipse detection algorithm based on pseudo-random sample consensus. In *Computer Analysis of Images and Patterns*, vol. 4673 of *Lecture Notes in Computer Science*. Springer Berlin / Heidelberg, 2007, pp. 669–676.
- [167] SRESTASATHIERN, P., AND YILMAZ, A. Planar shape representation and matching under projective transformation. *Computer Vision and Image Understanding* 115, 11 (2011), 1525 – 1535.
- [168] STURM, P., AND MAYBANK, S. On plane-based camera calibration: A general algorithm, singularities, applications. In *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.* (1999), vol. 1, pp. –437 Vol. 1.
- [169] STURM, P., AND RAMALINGAM, S. A generic calibration concept: Theory and algorithms. Technical Report 5058, INRIA, dec 2003.
- [170] STURM, P., AND RAMALINGAM, S. A generic concept for camera calibration. In *Proc. European Conference on Computer Vision* (May 2004), vol. 2, Springer, pp. 1–13.
- [171] SWAMINATHAN, R., AND NAYAR, S. K. Nonmetric calibration of wide-angle lenses and polycameras. *IEEE Trans. Pattern Anal. Mach. Intell.* 22, 10 (Oct. 2000), 1172–1178.
- [172] SWARNINATHAN, R., AND NAYAR, S. Non-metric calibration of wide-angle lenses and polycameras. In *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.* (1999), vol. 2, pp. –419 Vol. 2.
- [173] TAN, R. T., NISHINO, K., AND IKEUCHI, K. Separating reflection components based on chromaticity and noise analysis. *IEEE TPAMI* 26, 10 (2004), 1373–1379.
- [174] TARDIF, J.-P., STURM, P., AND ROY, S. Self-calibration of a general radially symmetric distortion model. In *Computer Vision – ECCV 2006*, A. Leonardis, H. Bischof, and A. Pinz, Eds., vol. 3954 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2006, pp. 186–199.
- [175] TEIXEIRA, L., LOAIZA, M., RAPOSO, A., AND GATTASS, M. Augmented reality using projective invariant patterns. In *Advances in Visual Computing*, vol. 5358 of *Lecture Notes in Computer Science*. Springer Berlin / Heidelberg, 2008.
- [176] THORMÄHLEN, T., AND BROZIO, H. Automatic line-based estimation of radial lens distortion. *Integr. Comput.-Aided Eng.* 12, 2 (Apr. 2005), 177–190.
- [177] TOMINANGA, S., AND WANDELL, B. A. Standard surface-reflectance model and illuminant estimation. *Journal of the Optical Society of America A*, 6 (1989), 576–584.

- [178] TOSCANI, G. *Systemes de calibration et perception du mouvement en vision artificielle*. PhD thesis, 1987. Thèse de doctorat dirigée par Faugeras, Olivier Sciences appliquées Paris 11 1987.
- [179] TSAI, R. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal on Robotics and Automation* 3, 4 (1987), 323–344.
- [180] TSONISP, V. S., CH, K. V., AND TRAHANIASLJ, P. E. Landmark-based navigation using projective invariants. In *Proceedings of the 1998 IEEE Intl. Conf. on Intelligent Robots and Systems* (Victoria, Canada, 1998), IEEE Computer Society.
- [181] UCHIYAMA, H., AND SAITO, H. Random dot markers. *Virtual Reality Conference, IEEE 0* (2011), 271–272.
- [182] USABIAGA, J., EROL, A., BEBIS, G., BOYLE, R., AND TWOMBLY, X. Global hand pose estimation by multiple camera ellipse tracking. *Machine Vision and Applications* 21, 1 (2009), 1–15.
- [183] VAN RHIJN, A., AND MULDER, J. D. Optical tracking using line pencil fiducials. In *Proceedings of the eurographics symposium on virtual environments* (2004).
- [184] WAGNER, D., LANGLOTZ, T., AND SCHMALSTIEG, D. Robust and unobtrusive marker tracking on mobile phones. In *Proceedings of the 7th IEEE/ACM International Symposium on Mixed and Augmented Reality* (Washington, DC, USA, 2008), ISMAR '08, IEEE Computer Society, pp. 121–124.
- [185] WAGNER, D., REITMAYR, G., MULLONI, A., DRUMMOND, T., AND SCHMALSTIEG, D. Real time detection and tracking for augmented reality on mobile phones. *IEEE Transactions on Visualization and Computer Graphics* 99 (2010).
- [186] WALKER, M. W., SHAO, L., AND VOLZ, R. A. Estimating 3-d location parameters using dual number quaternions. *CVGIP: Image Underst.* 54, 3 (Oct. 1991), 358–367.
- [187] WALTHELM, A., AND KLUTHE, R. Active distance measurement based on robust artificial markers as a building block for a service robot architecture. In *IFAC Symposium on Artificial Intelligence in Real Time Control* (Budapest, 2000), Budapest Polytechnic.
- [188] WANG, J., SHI, F., ZHANG, J., AND LIU, Y. A new calibration model of camera lens distortion. *Pattern Recognition* 41, 2 (2008), 607 – 615.
- [189] WEBER, J., AND MALIK, J. Rigid body segmentation and shape description from dense optical flow under weak perspective. In *Computer Vision, 1995. Proceedings., Fifth International Conference on* (Jun 1995), pp. 251–256.

- [190] WEDEL, A., POCK, T., ZACH, C., BISCHOF, H., AND CREMERS, D. An improved algorithm for tv-l 1 optical flow. In *Statistical and Geometrical Approaches to Visual Motion Analysis*, D. Cremers, B. Rosenhahn, A. Yuille, and F. Schmidt, Eds., vol. 5604 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2009, pp. 23–45.
- [191] WENG, J., COHEN, P., AND HERNIOU, M. Camera calibration with distortion models and accuracy evaluation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 14, 10 (Oct 1992), 965–980.
- [192] WERLBERGER, M., POCK, T., AND BISCHOF, H. Motion estimation with non-local total variation regularization. In *CVPR (2010)*, pp. 2464–2471.
- [193] WRIGHT, J., WAGNER, A., RAO, S., AND MA, Y. Homography from coplanar ellipses with application to forensic blood splatter reconstruction. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on* (june 2006), vol. 1, pp. 1250 – 1257.
- [194] XIE, Y., AND JI, Q. A new efficient ellipse detection method. In *Pattern Recognition, 2002. Proceedings. 16th International Conference on* (2002), vol. 2, pp. 957–960 vol.2.
- [195] XIE, Y., AND OHYA, J. Efficient detection of ellipses from an image by a guided modified ransac. In *Image Processing: Algorithms and Systems* (2009), vol. 7245 of *SPIE Proceedings*, SPIE.
- [196] XU, L., CHEN, J., AND JIA, J. A segmentation based variational model for accurate optical flow estimation. In *ECCV (2008)*, Springer, pp. 671–684.
- [197] YAMAZAKI, S., MOCHIMARU, M., AND KANADE, T. Simultaneous self-calibration of a projector and a camera using structured light. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2011 IEEE Computer Society Conference on* (2011), pp. 60–67.
- [198] YING, X., AND ZHA, H. Camera calibration from a circle and a coplanar point at infinity with applications to sports scenes analyses. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on* (2007), pp. 220–225.
- [199] YOON, J.-H., PARK, J.-S., AND KIM, C. Increasing camera pose estimation accuracy using multiple markers. In *Advances in Artificial Reality and Tele-Existence*, vol. 4282 of *Lecture Notes in Computer Science*. Springer Berlin / Heidelberg, 2006, pp. 239–248.
- [200] YOON, Y., DESOUZA, G., AND KAK, A. Real-time tracking and pose estimation for industrial objects using geometric features. In *Robotics and Automation*,

2003. *Proceedings. ICRA '03. IEEE International Conference on* (sept. 2003), vol. 3, pp. 3473 – 3478 vol.3.
- [201] YOUSEF B. MAHDY, K. F. H., AND ABDEL-MAJID, M. A. Projector calibration using passive stereo and triangulation. *International Journal of Future Computer and Communication* 2, 5 (2013), 385 – 390.
- [202] YU, Q., LI, Q., AND DENG, Z. Online motion capture marker labeling for multiple interacting articulated targets. *Computer Graphics Forum* 26, 3 (2007), 477–483.
- [203] YU, R., YANG, T., ZHENG, J., AND ZHANG, X. Real-time camera pose estimation based on multiple planar markers. In *Proceedings of the 2009 Fifth International Conference on Image and Graphics* (Washington, DC, USA, 2009), ICIG '09, IEEE Computer Society, pp. 640–645.
- [204] YU, X., LEONG, H. W., XU, C., AND TIAN, Q. A robust and accumulator-free ellipse hough transform. In *Proceedings of the 12th annual ACM international conference on Multimedia* (New York, NY, USA, 2004), ACM.
- [205] ZHANG, X., FRONZ, S., AND NAVAB, N. Visual marker detection and decoding in ar systems: A comparative study. In *Proc. of the 1st International Symposium on Mixed and Augmented Reality* (Washington, DC, USA, 2002), ISMAR '02, IEEE Computer Society, pp. 97–.
- [206] ZHANG, Z. A flexible new technique for camera calibration. *IEEE Trans. Pattern Anal. Mach. Intell.* 22, 11 (Nov. 2000), 1330–1334.
- [207] ZHANG, Z. Camera calibration with one-dimensional objects. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 26, 7 (July 2004), 892–899.
- [208] ZICKLER, T., MALICK, S. P., KRIEGMAN, D. J., AND BELHUMEUR, P. N. Color subspaces as photometric invariants. *IJCVs* 79, 1 (2008), 13–30.
- [209] ZIMMER, H., BRUHN, A., WEICKERT, J., VALGAERTS, L., SALGADO, A., ROSENHAHN, B., AND SEIDEL, H.-P. Complementary optic flow. In *Energy Minimization Methods in Computer Vision and Pattern Recognition*, D. Cremers, Y. Boykov, A. Blake, and F. Schmidt, Eds., vol. 5681 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2009, pp. 207–220.



Università
Ca' Foscari
Venezia

DEPOSITO ELETTRONICO DELLA TESI DI DOTTORATO

DICHIARAZIONE SOSTITUTIVA DELL'ATTO DI NOTORIETA'

(Art. 47 D.P.R. 445 del 28/12/2000 e relative modifiche)

Io sottoscritto ... FILIPPO BERGAMASCO

nat. a ... VERONA (prov. VR.) il ... 18/09/1985

residente a ... MESTRE in ... VIA NAPOLI n. 45

Matricola (se posseduta) ... 820576 Autore della tesi di dottorato dal titolo:

... HIGH-ACCURACY CAMERA CALIBRATION AND SCENE
... ACQUISITION

Dottorato di ricerca in ... INFORMATICA

(in cotutela con

Ciclo ... 27

Anno di conseguimento del titolo ... 2015

DICHIARO

di essere a conoscenza:

- 1) del fatto che in caso di dichiarazioni mendaci, oltre alle sanzioni previste dal codice penale e dalle Leggi speciali per l'ipotesi di falsità in atti ed uso di atti falsi, decado fin dall'inizio e senza necessità di nessuna formalità dai benefici conseguenti al provvedimento emanato sulla base di tali dichiarazioni;
- 2) dell'obbligo per l'Università di provvedere, per via telematica, al deposito di legge delle tesi di dottorato presso le Biblioteche Nazionali Centrali di Roma e di Firenze al fine di assicurarne la conservazione e la consultabilità da parte di terzi;
- 3) che l'Università si riserva i diritti di riproduzione per scopi didattici, con citazione della fonte;
- 4) del fatto che il testo integrale della tesi di dottorato di cui alla presente dichiarazione viene archiviato e reso consultabile via internet attraverso l'Archivio Istituzionale ad Accesso Aperto dell'Università Ca' Foscari, oltre che attraverso i cataloghi delle Biblioteche Nazionali Centrali di Roma e Firenze;
- 5) del fatto che, ai sensi e per gli effetti di cui al D.Lgs. n. 196/2003, i dati personali raccolti saranno trattati, anche con strumenti informatici, esclusivamente nell'ambito del procedimento per il quale la presentazione viene resa;
- 6) del fatto che la copia della tesi in formato elettronico depositato nell'Archivio Istituzionale ad Accesso Aperto è del tutto corrispondente alla tesi in formato cartaceo, controfirmata dal tutor, consegnata presso la segreteria didattica del dipartimento di riferimento del corso di dottorato ai fini del deposito presso l'Archivio di Ateneo, e che di conseguenza va esclusa qualsiasi responsabilità dell'Ateneo stesso per quanto riguarda eventuali errori, imprecisioni o omissioni nei contenuti della tesi;
- 7) del fatto che la copia consegnata in formato cartaceo, controfirmata dal tutor, depositata nell'Archivio di Ateneo, è l'unica alla quale farà riferimento l'Università per rilasciare, a richiesta, la dichiarazione di conformità di eventuali copie.

Data 27/11/2014

Firma Filippo Bergamasco

AUTORIZZO

- l'Università a riprodurre ai fini dell'immissione in rete e a comunicare al pubblico tramite servizio on line entro l'Archivio Istituzionale ad Accesso Aperto il testo integrale della tesi depositata;
- l'Università a consentire:
 - la riproduzione a fini personali e di ricerca, escludendo ogni utilizzo di carattere commerciale;
 - la citazione purché completa di tutti i dati bibliografici (nome e cognome dell'autore, titolo della tesi, relatore e correlatore, l'università, l'anno accademico e il numero delle pagine citate).

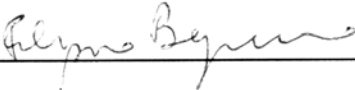
DICHIARO

- 1) che il contenuto e l'organizzazione della tesi è opera originale da me realizzata e non infrange in alcun modo il diritto d'autore né gli obblighi connessi alla salvaguardia di diritti morali od economici di altri autori o di altri aventi diritto, sia per testi, immagini, foto, tabelle, o altre parti di cui la tesi è composta, né compromette in alcun modo i diritti di terzi relativi alla sicurezza dei dati personali;
- 2) che la tesi di dottorato non è il risultato di attività rientranti nella normativa sulla proprietà industriale, non è stata prodotta nell'ambito di progetti finanziati da soggetti pubblici o privati con vincoli alla divulgazione dei risultati, non è oggetto di eventuale registrazione di tipo brevettuale o di tutela;
- 3) che pertanto l'Università è in ogni caso esente da responsabilità di qualsivoglia natura civile, amministrativa o penale e sarà tenuta indenne a qualsiasi richiesta o rivendicazione da parte di terzi.

A tal fine:

- dichiaro di aver autoarchiviato la copia integrale della tesi in formato elettronico nell'Archivio Istituzionale ad Accesso Aperto dell'Università Ca' Foscari;
- consegno la copia integrale della tesi in formato cartaceo presso la segreteria didattica del dipartimento di riferimento del corso di dottorato ai fini del deposito presso l'Archivio di Ateneo.

Data _____

Firma 

La presente dichiarazione è sottoscritta dall'interessato in presenza del dipendente addetto, ovvero sottoscritta e inviata, unitamente a copia fotostatica non autenticata di un documento di identità del dichiarante, all'ufficio competente via fax, ovvero tramite un incaricato, oppure a mezzo posta

Firma del dipendente addetto

Ai sensi dell'art. 13 del D.Lgs. n. 196/03 si informa che il titolare del trattamento dei dati forniti è l'Università Ca' Foscari - Venezia.

I dati sono acquisiti e trattati esclusivamente per l'espletamento delle finalità istituzionali d'Ateneo; l'eventuale rifiuto di fornire i propri dati personali potrebbe comportare il mancato espletamento degli adempimenti necessari e delle procedure amministrative di gestione delle carriere studenti. Sono comunque riconosciuti i diritti di cui all'art. 7 D. Lgs. n. 196/03.

Estratto per riassunto della tesi di dottorato

L'estratto (max. 1000 battute) deve essere redatto sia in lingua italiana che in lingua inglese e nella lingua straniera eventualmente indicata dal Collegio dei docenti.

L'estratto va firmato e rilegato come ultimo foglio della tesi.

Studente: Filippo Bergamasco _____ matricola: 820576 _____

Dottorato: Informatica _____

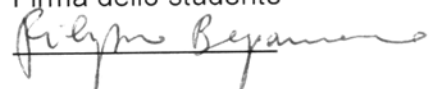
Ciclo: 27 _____

Titolo della tesi¹: High-Accuracy Camera Calibration and Scene Acquisition _____

Abstract:

In this thesis we present some novel approaches in the field of camera calibration and high-accuracy scene acquisition. The first part is devoted to the camera calibration problem exploiting targets composed by circular features. Specifically, we start by improving some previous work on a family of fiducial markers which are leveraged to be used as calibration targets to recover both extrinsic and intrinsic camera parameters. Then, by using the same geometric concepts developed for the markers, we present a method to calibrate a pinhole camera by observing a set of generic coplanar circles. In the second part we move our attention to unconstrained (non-pinhole) camera models. We begin asking ourselves if such models can be effectively applied also to quasi-central cameras and present a powerful calibration technique that exploit active targets to estimate the huge number of parameters required. Then, we apply a similar method to calibrate the projector of a structured-light system during the range-map acquisition process to improve both the accuracy and coverage. Finally, we propose a way to lower the complexity of a complete unconstrained model toward a pinhole configuration but allowing a complete generic distortion map. In the last part we study two different scene acquisition problems, namely: Accurate 3D shape reconstruction and object material recovery. In the former, we propose a novel visual-inspection device for the dimensional assessment of metallic pipe intakes. In the latter, we formulate a total-variation regularized optimization approach for the simultaneous recovery of the optical flow and the dichromatic coefficients of a scene by analyzing two subsequent frames.

Firma dello studente



¹ Il titolo deve essere quello definitivo, uguale a quello che risulta stampato sulla copertina dell'elaborato consegnato.